The Dissertation Committee for Robert C. Daniels
certifies that this is the approved version of the following dissertation:

Machine Learning for Link Adaptation in Wireless Networks

Committee:

Robert W. Heath, Jr., Supervisor

Jeffrey Andrews

Scott Nettles

Constantine Caramanis

Lili Qiu

Machine Learning for Link Adaptation in Wireless Networks

by

Robert C. Daniels, B.S.; M.S.E.

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

December 2011

Dedicated to my son, William.

Acknowledgments

I wish to thank all of those who have inspired, enlightened, and challenged me during my tenure at the University of Texas. This includes my colleagues in the WNCG and WSIL as well as my advisor Professor Robert W. Heath, Jr. I especially want to thank Professor Heath for his patience, in particular for tolerating my stubbornness on selecting research topics. I would also like to thank my committee members Professor Jeffrey G. Andrews, Professor Constantine M. Caramanis, Professor Scott M. Nettles, and Professor Lili Qiu for their challenging questions and guidance throughout this program. Finally, I would like to thank my family. My parents, Richard and Lynn Daniels, have always remained a positive force in my life no matter how disagreeable I've become over the years. My ultimate motivation is my son, William. I am certain that without William, I could not have found the courage to finish my studies.

Machine Learning for Link Adaptation in Wireless Networks

Publication No. _____

Robert C. Daniels, Ph.D.
The University of Texas at Austin, 2011

Supervisor: Robert W. Heath, Jr.

Link adaptation is an important component of contemporary wireless networks that require high spectral efficiency and service a variety of network applications/configurations. By exploiting information about the wireless channel, link adaptation strategically selects wireless communication transmission parameters in real-time to optimize performance. Link adaptation in practice has proven challenging due to impairments outside system models and analytical intractability in modern broadband networks with multiple antennas (MIMO), orthogonal frequency division multiplexing (OFDM), forward error correction, and bit-interleaving. The objective of this dissertation is to provide simple and flexible link adaptation algorithms with few link model assumptions that are amenable to modern wireless networks.

First, a complete design and analysis of supervised learning for link adaptation in MIMO-OFDM is provided. This includes the construction of

a publicly available data set, a new frame error rate bound leading to a new feature set, and IEEE 802.11n performance evaluation to verify that my design outperforms existing link quality metrics. Next, two supervised learning classification algorithms are designed to exploit information collected from packets transmitted and received over standard links in real time: database learning with nearest neighbor classifiers and support vector machines. Algorithms are also proposed to preserve diversity of feature sets in the database and to allow learning algorithms to seek out more information about the network.

Finally, link adaptation with supervised learning is applied to MIMO-OFDM systems where the modulation order may be adapted per-stream. This leads to the analysis of the ordered SNR per stream and its connection to the cumulative distribution function of SNR on each stream. Decoupled link adaptation algorithms, which significantly reduce the complexity of non-uniform link adaptation algorithms, are proposed. New analysis of non-uniform link adaptation shows that the performance of decoupled link adaptation algorithms converge to the performance of joint (optimal) link adaptation algorithms as the number of modulation and coding options per-stream increase. This guides the construction of future standards to reduce the complexity of link adaptation in MIMO-OFDM.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The role of the wireless network in the global information infrastructure is ever-expanding. In 1996, just over 16% of Americans owned a cellular telephone.[1] Today, the penetration rate is nearly 90% with the addition of new wireless markets such as wireless personal area networking (WPAN) and wireless local area networking (WLAN), whose device market has increased to 5.5 billion USD annually and is expected to continue to rise.[2][3] In just over a decade, digital wireless products have grown from a consumer novelty to a vital component of personal communication, entertainment, commerce, and industry.

Naturally, because of the significant role that digital wireless technology increasingly serves in our daily lives, maximizing the data rate, efficiency, and robustness of digital wireless networks is of critical importance. To this end, digital wireless communications research has a long history, approaching a century, of investigating three link constructs for various wireless network configurations: *capacity* or theoretical bounds on feasible and achievable data

[1] According to Mobile Phone News.
[2] According to the CWTA & the Dell-Oro Group.
[3] ABI Research, Inc. estimates that 90% of smartphones will incorporate WLAN by 2014.

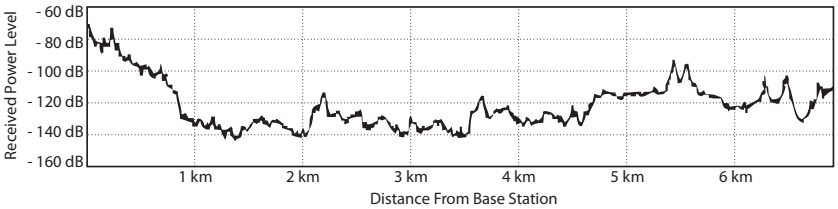
rates, *error rate* or the practical communication reliability over wireless links, and *overhead* or the excess control information that enables practical wireless communications [14, 16, 35]. In this dissertation I will investigate a topic in wireless research that relates all three of the aforementioned link constructs in a practical manner: link adaptation. I improve on practical link adaptation by exploiting a wealth of theory and a suite of algorithms from the field of machine learning.

1.1 Wireless Link Adaptation

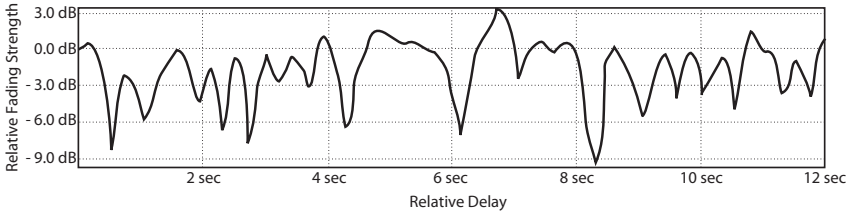
The capacity of a wireless link varies according to the observed wireless medium or *channel* [35]. In digital wireless communications, the gap between capacity and the practical link data rate (which depends on both error rate and overhead) for a fixed channel also varies according to the selected transmission parameters.[4] Because the channel changes very often within typical links, communication with fixed transmission parameters is inadvisable since the capacity gap is quite sensitive to the selected transmission parameters [14]. A mismatch of channel parameters and wireless channel state will either result in extremely conservative data rates or complete link failure [48]. Consequently, wireless systems implement *link adaptation* or the dynamic selection of transmission parameters for data transmitted over dynamic links.

Consider Figure 1.1 which typifies the channel in cellular communica-

[4]Example transmission parameters include the digital symbol constellation and the binary error control coding scheme. For more examples see Chapter 2 of this dissertation.



(a) Average path loss measurements in Tokyo at 457.2 MHz. This figure was recreated from data in [71] (this is an original figure).



(b) Small scale fading snapshot for 457.2 MHz at walking speeds [31].

Figure 1.1: Channel variability in wireless systems. Total received power (in dB) in wireless links is the sum of average path loss and small scale fading.

tion. In narrowband, noise-limited environments, the received power of wireless signals is directly related to the practical data rate through link adaptation (assuming fixed receiver noise properties, otherwise performance is dictated by the signal-to-noise ratio (SNR)). Without link adaptation, the transmission parameters must be designed according to the worst-case scenario to prevent complete link failure. In Figure 1.1 this would result in "throwing away" over 20 dB in received signal power on average. By exploiting information about the channel (in this case according to the received power), link adaptation maximizes the practical data rate. As a consequence, link adaptation is an important component of contemporary spectrum-limited wireless networks that require high data rates.

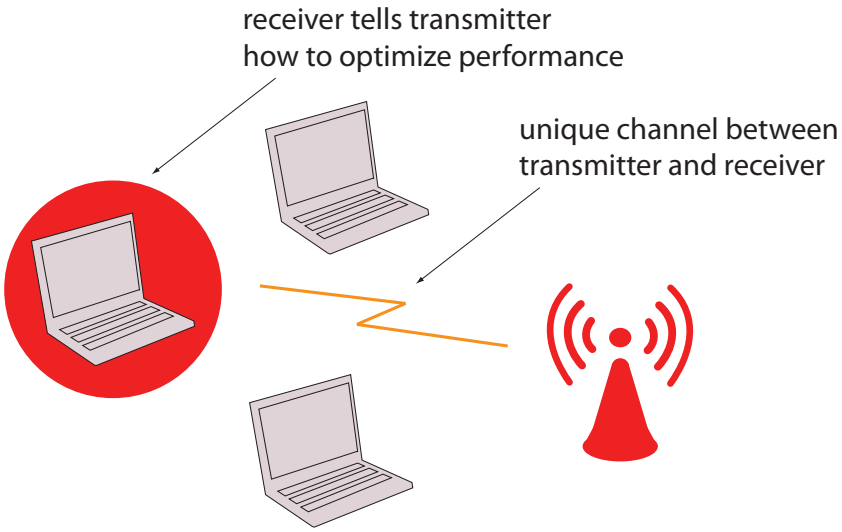


Figure 1.2: Basic illustration of fast link adaptation.

This dissertation focuses on the most popular form of link adaptation, *fast* link adaptation, which exploits instantaneous knowledge of channel information for instantaneous selection of transmission parameters [48]. Figure 1.2 demonstrates the basic concept of fast link adaptation in the WLAN environment (although the basic procedure is essentially the same for all wireless links). The receiver (laptop) observes the wireless channel to the transmitter (base station) and informs the base station which transmission parameters to select to maximize data rate, assuming that the same channel will be observed in future transmissions.

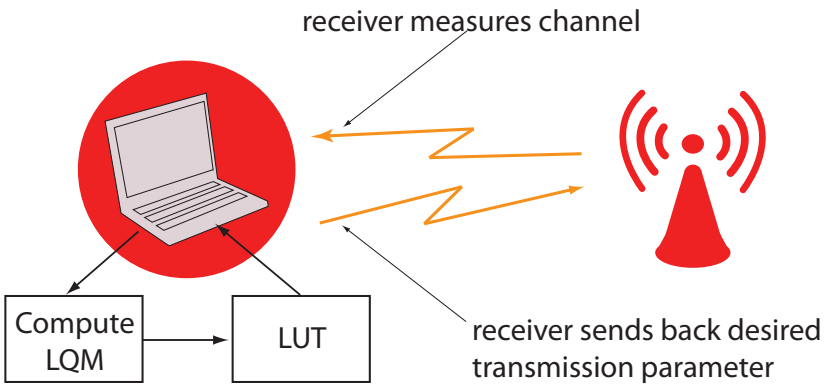


Figure 1.3: Basic illustration of fast link adaptation with look-up-tables.

1.2 Illustrating Current Link Adaptation Deficiencies

Currently, practical fast link adaptation algorithms assume a very simple and straightforward procedure as illustrated in Figure 1.3. First, the receiver measures the wireless channel and computes a *link quality metric* (LQM). Second, the link quality metric is compared to a look-up-table (LUT) that resides at the receiver. Finally, the transmission parameters that match the link quality metric entry in the look-up-table are forwarded to the transmitter. This procedure is limited in two primary ways. First, link quality metrics must be of extremely small dimension (typically 1- or 2-dimensional [48]) so that they are amenable to look-up-tables. Single-dimensional link quality metrics can always be enabled through the derivation of closed-form expressions of the practical data rate as a function of the wireless channel. In the past this was straightforward for narrowband wireless communication systems that only featured a single antenna at the transmitter and receiver. Modern wireless networks, however, feature wideband links with orthogonal frequency division

multiplexing (OFDM), multiple antenna devices with multiple-input multiple-output (MIMO) processing, and bit-interleaving combined with error control coding over spatial and frequency selective channels. Due to the complexity of these systems, only closed form *approximations* of the data rate are currently available [46, 56, 65, 75, 88]. Without exact closed-form expressions and without flexibility with more complicated link quality metrics, link adaptation accuracy has been compromised [8, 9, 12, 25, 48, 52, 53, 87, 94]. Consider Figure 1.4, which compares look-up-table based link adaptation to machine learning link adaptation procedures developed in this dissertation. The link adaptation procedures that I develop through machine learning provide flexibility with more complicated multi-dimensional link metrics that more accurately characterize the wireless channel, leading to enhanced data rates.

The second limitation of the link adaptation procedure in Figure 1.3 is that look-up-tables are created offline, before device operation. Unfortunately, it is very difficult to capture all potential system effects without real-time measurements. Practical transceivers exhibit nonlinearities in a multitude of hardware components including power and low-noise amplifiers, mixers, and analog filters [27, 99, 108]. Further, finite resolution in digital-to-analog/analog-to-digital (DAC/ADC) converters, local oscillator instability (phase noise), and ambient interference all contribute additive noise with different statistical distributions [15, 23]. Perhaps most troubling is that nonlinearities and non-Gaussian noise effects vary for each wireless device. Moreover, these effects may evolve throughout the lifetime of a wireless link according to environmen-

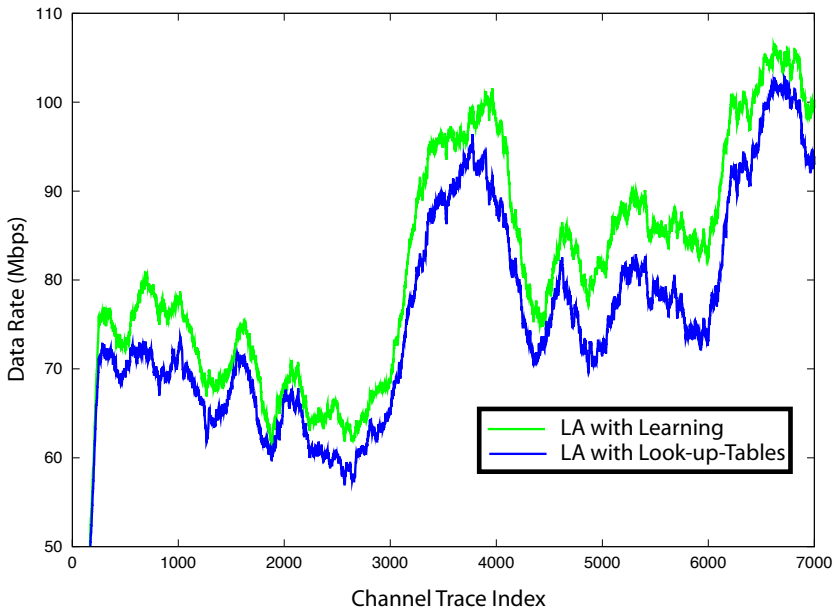


Figure 1.4: Trace of 7000 example channels demonstrating the advantage of link adaptation through machine learning with more accurate link quality metrics in IEEE 802.11n WLAN with 2 transmit antennas and 2 receive antennas.

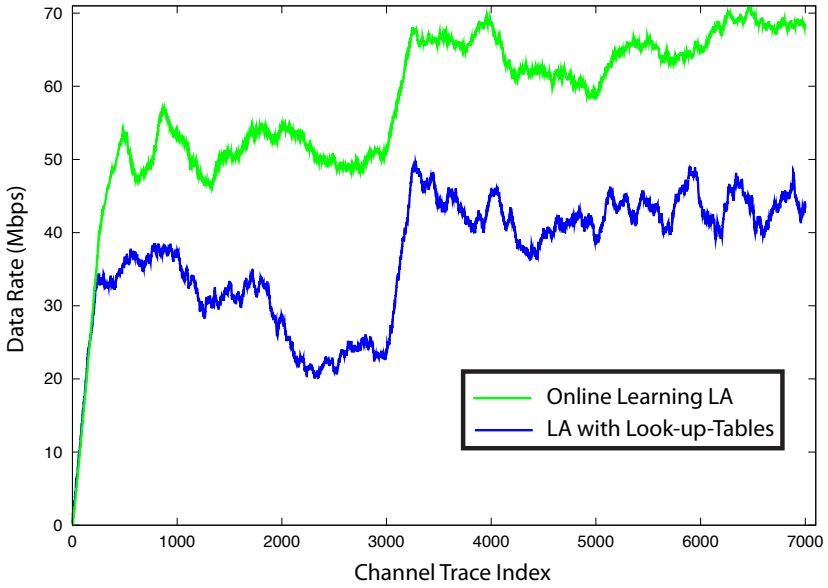


Figure 1.5: Trace of 7000 example channels demonstrating the advantage of online learning to compensate for unpredicted nonlinearities in IEEE 802.11n WLAN with 2 transmit antennas and 2 receive antennas.

tal factors such as device temperature and co-channel interference. I have first hand experience on the IEEE 802.11n prototype, Hydra [62], that shows the performance of an offline link adaptation algorithm is unique to each transmit/receive device pair even in the absence of interference. MIT's Roofnet project observed similar effects with commercial IEEE 802.11 transceivers [1]. This result is not surprising since die-process variations in high-yield chip manufacturing can be significant. As the technology shrinks in dimension, variations will become increasingly problematic [70]. As a consequence, the performance of link adaptation based on offline analysis, including look-up-tables, can be very suboptimal.

Consider Figure 1.5, which adds solid-state power amplifier nonlinearity (without sufficient backoff) to the same channels featured in Figure 1.4 [99]. The performance of link adaptation with look-up-tables suffers tremendously, partially due to the negative effect that nonlinearities have on system performance. Much of the loss, however, is due to the incorrect selection of transmission parameters because the look-up-table was created assuming the absence of channel nonlinearities. In this dissertation I enable smarter link adaptation through machine learning algorithms that update in real-time to adjust to any changing system conditions of any transmit/receive device pair. Figure 1.5 shows that the advantage of these online learning algorithms in systems impaired by unmodeled channel effects can be significant.

1.3 Machine Learning

As displayed in the previous section through a performance preview of the high-level contributions in this dissertation, future link adaptation algorithms ideally feature a simple and flexible implementation that captures the complex nature of wireless link performance with few link model assumptions. This will save crucial engineering design time used to analyze the performance of all transmission parameters for all channel conditions including those impaired by nonlinearities and non-Gaussian additive noise effects. Further, deployed wireless devices will not have to adapt conservatively to avoid the deleterious effect of these impairments if they cannot be accurately modeled during offline system design or if they evolve during wireless network opera-

tion. As a path to this future, my dissertation provides a paradigm shift in link adaptation through the application of machine learning.

Machine learning facilitates a simple *and* general approach to link adaptation because machine learning algorithms only observe the link through a training set of past channel observations and their corresponding performance in terms of the transmission parameters. With machine learning, link adaptation can evolve its response without specific knowledge of the underlying system model or the hardware they will be implemented on. While machine learning algorithms are powerful, significant research was necessary to overcome fundamental limitations in their direct application. For example, machine learning algorithms generally assume that the system response will be favorable if the correct transmission parameters are selected. The wireless channel, however, is not deterministic and regardless of which transmission parameters are selected, failed communication is still possible. Machine learning algorithms also often assume that the system does not change over time, i.e., that link adaptation maintains a consistent mapping between link quality metrics and the best transmission parameters. As discussed in Section 1.2, however, nonlinear channel and non-Gaussian additive noise effects, may evolve over time. To complete link adaptation through machine learning, I had to overcome many obstacles.

Despite its compelling use for empirical inference, machine learning has not been historically utilized to a large extent in digital wireless communications for several reasons. First, communication and information theory

research based on the analysis of fixed stochastic models has been enormously successful in architecting modern wireless networks. Unsurprisingly, many members of this community are inclined to continue on this path. Second, proper application of machine learning to physical layer wireless design requires intimate knowledge of both wireless communication systems and machine learning theory. Until recently, their has not been significant crossover between these two communities. Third, it is preferable for wireless communication algorithms to maintain minimal implementation complexity. Machine learning has long been associated with practicality concerns due to the high complexity displayed by the baseline implementation of traditional machine learning algorithms. In this dissertation I provide a complete solution for link adaptation with machine learning, showing not only the performance advantage, but also its practicality.

1.4 Thesis Statement

Machine learning simultaneously improves the accuracy and simplifies the design of link adaptation in practical wireless networks through observation of past performance.

1.5 Contributions

The contributions of this dissertation are broadly separated into three categories as follows.

- **Supervised Learning for Link Adaptation**: I provide a complete design and analysis of supervised learning for link adaptation in practical MIMO-OFDM with coding and interleaving. This includes the construction of a publicly available offline data set to allow the academic community to compare the construction of feature sets and classifiers to optimize the accuracy of link adaptation. I construct my own feature set, based on a new frame error rate bound, that outperforms existing link quality metrics for OFDM. This work shows the potential of machine learning to deal with complexities in modern MIMO-OFDM systems and also provides the baseline performance of link adaptation with machine learning.

- **Online Link Adaptation**: Online link adaptation is necessary to tune link adaptation to the unique nonlinearities and non-Gaussian noise effects of each transmit/receive device pair. I accomplish this by designing supervised learning classification algorithms that exploit information collected from packets transmitted and received over standard links. I create a database learning (online learning) system based on nearest neighbor classifiers that both preserve diversity in the observations and allows the learning to evolve over time. Moreover algorithms for exploration are provided to allow learning algorithms to seek out more information about the network. Measurements of a MIMO-OFDM prototype validate the efficacy of the proposed online database learning system. Because database learning requires significant memory storage

and processing complexity, I create online learning through support vector machines which drastically reduce the complexity of online learning while maintaining accuracy in link adaptation performance.

- **Link Adaptation in MIMO-OFDM with Non-Uniform Streams**: I extend link adaptation to coded and interleaved MIMO-OFDM systems where the modulation order may be adapted per-stream, which has not yet been studied significantly outside of wireless standard contributions. The main focus of this work is on reducing the complexity of link adaptation with non-uniform streams. My analysis of the ordered SNR per stream discovers a connection to the inverse marginal cumulative distribution function of SNR on each stream. This discovery is exploited to construct new models of the channel and hence for new feature sets with reduced dimension. Further, decoupled link adaptation algorithms, which significantly reduce the complexity of non-uniform link adaptation algorithms, are proposed. New analysis of non-uniform link adaptation shows that the performance of decoupled link adaptation algorithms converge joint (optimal) link adaptation algorithms as the number of modulation and coding options per-stream increase. This guides the construction of future standards to reduce the complexity of link adaptation in MIMO-OFDM.

1.6 Organization

Chapter 2 provides a general summary of link adaptation in wireless systems as well as the MIMO-OFDM system model considered in this dissertation. The MIMO-OFDM dataset constructed for this dissertation is also summarized. Chapter 3 designs a procedure for supervised learning classification in an offline manner, demonstrated with nearest neighbor classifiers, assuming the standard link model. Feature sets are designed and validated for MIMO-OFDM systems. Chapter 4 designs supervised learning algorithms for link adaptation, based on nearest neighbor and support vector machines, that operate in an online manner. Chapter 5 extends supervised learning link adaptation to MIMO-OFDM models with per-stream modulation and coding. Reduced complexity link adaptation procedures are also provided. Finally, Chapter 6 provides conclusions for this dissertation with a view towards future work.

1.7 Notation

Vectors are always assumed as column vectors unless otherwise specified and written as bold lowercase letters, $\mathbf{a}$. Variables, a, after Fourier domain transformation, are capitalized and written with Sans Serif fonts, A. Matrices and vectors of Fourier transformed variables may both be written as bold Sans Serif captial letters, $\mathbf{A}$, so care has been taken to always introduce its dimension clearly when defined. Matrices are written with bold uppercase letters, $\mathbf{A}$. $[\mathbf{A}]_{a,b}$ is the matrix element in the ath row and bth column. $[\mathbf{A}]_{a_1:a_2,b_1,b_2}$ denotes

the submatrix of $\mathbf{A}$ containing rows $a_1 \rightarrow a_2$ and columns $b_1 \rightarrow b_2$ while $[\mathbf{A}]_{:,b_1:b_2}$ is the submatrix containing all rows and columns $b_1 \rightarrow b_2$. $(\cdot)^*$ is the Hermitian transpose which defaults to standard transpose for real valued vectors and matrices. $\|\mathbf{A}\|_F$ denotes the Frobenius norm of a matrix $\mathbf{A}$. The trace of a matrix is denoted by $\mathsf{tr}(\mathbf{A})$. The conjugate of all elements in a vector or matrix, without transposition, is indicated with $\mathsf{conj}(\cdot)$. I represent a sequence with index k from k_1 to k_2 as $\{\cdot\}_{k=k_1}^{k_2}$. Circularly symmetric complex Gaussian vectors with mean $\boldsymbol{\mu}$ and covariance $\mathbf{R}$ are indicated by $\sim \mathcal{CN}(\boldsymbol{\mu}, \mathbf{R})$. $\mathbb{C}$ and $\mathbb{R}$ denote the field of complex and real numbers, respectively. $\mathbb{Z}$ is the set of integers and $\mathbb{N}$ is the set of non-negative integers.

Chapter 2

Background

This chapter provides prerequisite knowledge on link adaptation, summarizes the relevant system model for the remainder of the dissertation, and supplies a brief background on machine learning concepts.

2.1 Introduction

Link adaptation is critical to the operation of a digital communication system. The wireless medium, or wireless channel, exhibits highly variable conditions. Without link adaptation, wireless networks would have to be configured for the worst-case wireless channel conditions. Maintaining a sufficient reliability for all potential channel conditions would result in multiple orders of magnitude in lost data rate when favorable channel conditions are experienced. To further demonstrate this point, consider Table 2.1, which summarizes the data rates for current commercial wireless standards. Minimum rates are designed to satisfy maximum reliability, i.e, for bad channel conditions. Without link adaptation, the potential rate loss can be staggering.

Table 2.1: Minimum and maximum available rates for selected commercial wireless standards

Standard	**Max Rate**	**Min Rate**	**Lost Rate**
IEEE 802.11n[1]	600 Mbps	6.5 Mbps	593.5 Mbps
WiMax[2]	144 Mbps	14.4 Mbps	129.6 Mbps
3GPP-LTE[3]	326.4 Mbps	12.1 Mbps	314.3 Mbps
IEEE 802.15.3c[4]	5280 Mbps	25.8 Mbps	5254.2 Mbps

Link adaptation in this dissertation may refer to the configuration and/or selection of the following digital communication parameters before transmission of data.

- Digital symbol constellation – this represents the encapsulation of (potentially) multiple bits into a single symbol. This may implemented with, for example, quadrature amplitude modulation (QAM) or phase shift keying (PSK) of arbitrary dimension.

- Forward error correction (FEC) architecture and configuration – this represents the redundancy added to bits in order to protect against intermittent channel quality. FEC may be implemented with, for example, convolutional, Reed-Solomon, turbo, or low-density parity check (LDPC) codes with variable rate.

- Transmit signal power – this is limited by the spectrum regulations and

[1] 40 MHz bandwidth with 4 antennas at each end.
[2] Release 1.5 Downlink, 20 MHz bandwidth with 2 antennas at each end.
[3] Release 8 Downlink, 20 MHz bandwidth with 4 antennas at each end.
[4] SC PHY implementation.

the hardware constraints.

- Transmit signal bandwidth – this is also limited by the spectrum regulations and the hardware constraints.

- Operating frequency – the choice of the operating frequency determines properties of transmission including range as well as the licensing opportunities. Different operating frequencies also place different constraints on the hardware design.

- Pulse shaping – the transmitted waveform is impacted by digital and analog filters at the transmitter. This, in turn, influences the performance at the receiver as a function of the wireless channel.

- Modulation strategy – modulation strategies enable transmission over wireless channels to (potentially) multiple users.

  - OFDM parameters (e.g., number of subcarriers, cyclic prefix size, subcarrier energy distribution, etc.)

  - Spread spectrum parameters (e.g., spreading sequence and length)

- Multiple antenna parameters

  - Number of spatial streams – multiple antennas may be exploited to transmit several independent data streams in parallel.

  - Precoding strategy – precoding improves the performance of multi-antenna transmission when more than one spatial stream is used.

Examples include singular value decomposition (SVD) and limited feedback precoding.

- Diversity strategy – diversity transmission improves the performance of multiple-antenna transmission even when only a single spatial stream is used. Examples include space-time block coding and beamforming.
- Antenna selection – antennas can be selected based on the channel quality they observe.

In most wireless communication networks, particularly those based on standards, many of these parameters are already selected and/or configured based on the application they are intended to serve. For example, the transmit signal bandwidth is usually fixed based on available spectrum and the data rate required. Moreover, the operating frequency is usually selected based on the distance of typical links and the availability of spectrum. The parameters usually available to link adaptation are the digital symbol constellation, the FEC configuration, transmit signal energy, modulation configuration, the number of spatial streams, and the precoding/diversity strategy to use.

2.2 Fast Link Adaptation

The information that is known about the wireless medium or *side information*, is critical to the link adaptation process. Fast link adaptation is the instantaneous selection of communication parameters based on instantaneous

knowledge of the wireless channel state. Because of this, fast link adaptation assumes the slow fading model which states that the duration that the wireless medium remains static is larger than the transmission time of a physical layer frame. Most current wireless systems feature fast link adaptation since under ideal conditions (perfect wireless channel state knowledge, zero-delay feedback, etc) it maximizes the data rate of a wireless communication systems with a fixed selection of parameters. The slow fading assumption is not justified whenever there is significant mobility of the transmitter, receiver, or objects in the wireless environment, such as is the case in automobiles. Therefore, fast link adaptation is typically valid for moderate and low Doppler outdoor networks or indoor networks [77].

2.2.1 ARQ and Feedback Issues

Communication, wireless or otherwise, may be broken down into layers as shown in Table 2.2. Although link adaptation is mostly concerned with the operation of the physical layer (PHY), link adaptation is fundamentally a cross-layer process.

In this section I discuss how side information is exchanged between the transmitter and receiver in a network with automatic repeat request (ARQ), a mechanism for maintaining the integrity of data in the medium access control (MAC) layer. I assume that application-generating traffic resides on the TX Node, short for transmitting node, and the desired traffic destination is the RX Node, short for receiving node. This paradigm generalizes to appli-

Table 2.2: Layers in the communication stack

Layer	**Function**	**Level**
Application	process-to-process communications	7
Presentation	delivery/formatting of application information	6
Session	request/response handling	5
Transport	end-to-end connections; flow/congestion control	4
Network	routing and logical addressing	3
Logical Link Control	multiplexing/decoding MAC layer protocols	2(b)
Medium Access Control	device addressing and channel access control	2(a)
Physical	waveform creation, transmission, detection	1

cations that require multiple hops on a wireless network between the source generating traffic and the traffic destination. Multihop wireless links require that a link adaptation procedure is completed between each intermediate PHY transmit/receive pair (TX/RX Node).

The sequential procedure for communicating binary data from an application in the TX Node to an application in the RX Node using ARQ is as follows.

1. The application passes data bits down the communication stack to the medium access control (MAC) layer.

2. The MAC layer in the TX Node formats the bits it receives by affixing address information, MAC control information, and a cyclic redundancy check (CRC) to associate with the MAC layer in the RX Node.

3. Link adaptation exploits information collected by the MAC of the TX

Node to parametrize the PHY of the TX node such that data rate is maximized given reliability constraints imposed by the higher layers for the wireless link between the TX and RX node.

4. The MAC in the TX Node sends the PHY in the TX Node the MAC-formatted binary data.

5. The PHY in the TX Node translates the MAC-formatted binary data into physical waveforms that are transmitted over the wireless channel. The manner in which the binary data is formatted into a physical waveform determines the reliability and data rate of the wireless link. This data formatting depends explicitly on feedback from the RX node obtained by the MAC.

6. The PHY in the RX Node attempts to decipher the formatted data it receives. An estimate of the binary data created by the MAC at the TX Node is computed. The estimated binary data is passed up to the MAC at the RX Node.

7. Using the CRC, the MAC at the RX Node determines if any errors occur during the PHY transmission between the TX and RX Node. The next step depends on whether or not the CRC passes or fails.

   (a) If the CRC passes, the MAC at the RX Node generates a binary acknowledgment message (ACK).

(b) If the CRC fails, the MAC at the RX Node generates a binary negative acknowledgment message (NACK).

8. The (N)ACK message is formatted into a physical waveform using the PHY at the RX Node and transmitted over the wireless medium.[1]

9. The PHY at the TX Node receives the (N)ACK waveform and provides the binary representation of the (N)ACK message to the MAC at the TX Node.

10. If the ACK is received at the TX Node, the transmission of the MAC frame is successful. If a NACK message is received, the MAC retransmits by repeating steps (4) through (9).[2] Typically a MAC retransmission counter determines the maximum number of retransmissions before the MAC at the TX Node informs the higher layers at the TX Node that transmission was unsuccessful.

A timing diagram of the MAC ARQ procedure is shown in Figure 2.1. Figure 2.1 does not include the optional request-to-send (RTS), clear-to-send

[1]For the purposes of this dissertation I will always assume ACK messages are received successfully. This is usually a valid assumption since they are more robust than data messages by design. NACK messages may or may not successfully received, however, if the quality of the channel has degraded to the point for which no communication is possible over the wireless medium. In some systems, such as the IEEE 802.11 MAC, no explicit NACK message is sent, but instead NACK is determined by the TX Node if a timeout counter expires before the ACK message is received.

[2]In hybrid ARQ systems, the entire data packet is not retransmitted, but instead additional bits are sent to help correct errors that occur during the RX Node decoding operation. Hybrid ARQ is compatible with link adaptation principles discussed here, although it is not explicitly considered in this dissertation.

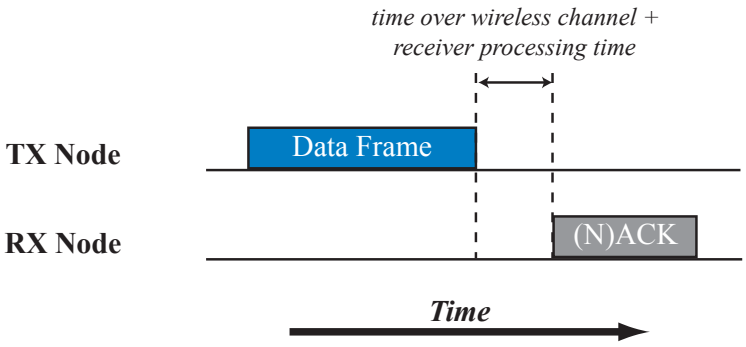


Figure 2.1: Automatic repeat request in the MAC

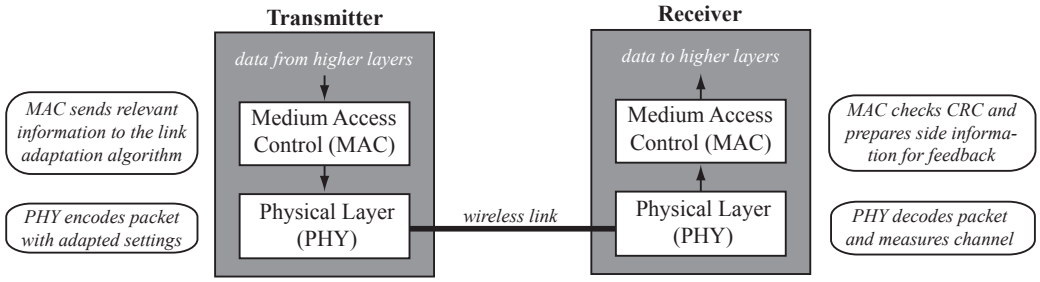


Figure 2.2: MAC/PHY processing summary to enable link adaptation.

(CTS) procedure in random access MACs (such as the DCF mode in IEEE 802.11) which is used to contend for access to the wireless medium and resolve the hidden node problem [95]. Many wireless networks centrally schedule multiple access (e.g. cellular networks) and will not need the RTS/CTS procedure. Figure 2.2 summarizes the MAC/PHY architecture for link adaptation.

Before transmission of any frame in the PHY, the MAC sends information it has collected through feedback from the RX node to the link adaptation algorithm at the transmitter in order to specify the communication parameters. Hence, feedback is necessary to enable link adaptation, unless reciprocity

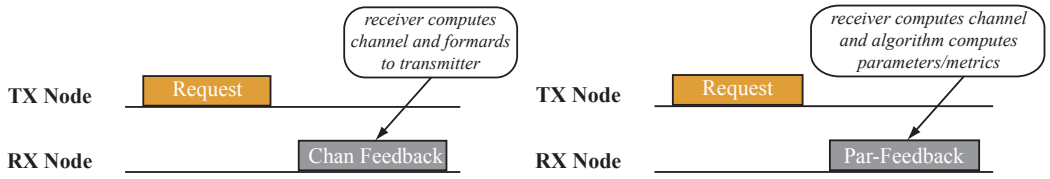


Figure 2.3: Two MAC feedback procedures to enable link adaptation. Feedback is either in parameter/link quality metric form or impulse response form.

is available as in some time division duplexed systems.[3] Link adaptation using wireless channel side information can be accomplished through feedback in two ways, as displayed in Figure 2.3. First, feedback of the channel impulse response can be requested from the receiver. Unfortunately, this requires a significant amount of bandwidth overhead. A more common practice is to make link adaptation decisions at the receiver and only feed back the recommended PHY parameters to the transmitter (which are referred to as the modulation and coding scheme (MCS) in Fig. 2.3). This reduces the feedback bandwidth. In IEEE 802.11n, for example, this reduces the feedback from hundreds or thousands of bits to less than 10.

2.2.2 Single Carrier Modulation with Frequency-Flat Channels

I begin by reviewing the foundations of link adaptation research. Early wireless systems operated over modest spectrum bandwidth due to hardware limitations and because wireless networks at that time largely carried

[3]Reciprocity is often difficult to implement in practice since the characteristics of the hardware are often not equivalent in both transmission directions.

voice traffic. Hence, this research focused on narrowband wireless links which are characterized by frequency-flat channels. Several independent discoveries showed that the symbol transmission rate [10], the transmitted signal energy [40], and the constellation order [98] could all be adapted as a function of the channel quality to maximize the rate subject to specified reliability constraints. It was not until the work of Goldsmith and Varaiya that link adaptation was formalized through a capacity optimization perspective and for variable dimension QAM constellations [33, 34]. In this section I demonstrate the practical consequences of this fundamental work.

Consider frame-based transmission of N_f bits, $\{b[i]\}_{i=0}^{N_f-1}$, in a single-input, single-output (SISO) digital communication system with slow fading over a bandwidth B Hz (note that B must be much less than the coherence bandwidth of the channel for narrowband assumptions). The N_f bits are transformed into digital symbols $\{x[n]\}_{n=0}^{N_f/K-1}$ through constellation $\mathcal{S} \subset \mathbb{C}$ with $|\mathcal{S}| = 2^K$. Let the transmit signal energy be E_s, the channel gain $h \in \mathbb{C}$, and the normalized additive noise $v[n] \in \mathbb{C}$ where $v[n] \sim \mathcal{NC}(0, \sigma^2)$, $\sigma = \sqrt{N_0 B}$, and $N_0/2$ is the noise density. Therefore, for input discrete-time data symbol $x[n]$ and output symbol $y[n]$ I have the relationship

$$y[n] = \sqrt{E_s}x[n]h + v[n] \tag{2.1}$$

where the channel gain is static for the entire frame through the slow fading assumption. At the receiver, received symbols are transformed into estimates of the original bit sequence, i.e., $\{y[n]\}_{n=0}^{N_f/K-1} \rightarrow \{\hat{b}[n]\}_{n=0}^{N_f-1}$. As mentioned

earlier, the goal of link adaptation is to maximize the rate while satisfying some reliability constraint. The commonly observed reliability constraint in frame-based transmission is the frame-error rate (FER). That is,

$$\mathrm{FER}(E_s, \sigma^2, h, N_f, M) = \Pr\left[\{b[n]\}_{n=0}^{N_f-1} \neq \{\hat{b}[n]\}_{n=0}^{N_f-1} | E_s, \sigma^2, h\right]. \qquad (2.2)$$

Stochastic constraints are necessary because additive noise realizations for each digital communication symbol cannot be determined before packet transmission. Higher layers use FER constraints to determine their own parameters including the number of retransmissions [44], the length of frames [38], and the network performance.

To evaluate FER generally, the bit-error rate metric P_B is invoked,

$$\mathrm{FER}(E_s, \sigma^2, h, N_f, \mathcal{S}) = 1 - \left(1 - P_B(E_s, \sigma^2, h, \mathcal{S})\right)^{N_f} \qquad (2.3)$$

where I have assumed each bit that is mapped to a constellation point in $\mathcal{S}$ exhibits equal probability of bit-error (e.g. Gray coding). It may be observed that, for narrow band systems in slow fading, the ratio $\gamma = E_s|h|/\sigma^2$ (SNR) provides a injective mapping to $P_B(\cdot)$ since E_s, h, and $1/\sigma^2$ proportionally contribute to signal strength for additive noise with normalized variance. With this system model, the link quality is completely determined by SNR and link adaptation in narrowband digital communication systems is completed by evaluating the SNR of the system. *Note: throughout this dissertation the signal-to-interference plus noise ratio (SINR) metric will not be considered explicitly. SINR is an important metric for cellular systems that are often*

more limited by interference than noise. I will operate under the assumptation that SINR is a special case of SNR with noise that is not necessarily distributed normally.

While the bandwidth, B, and the transmit signal energy, E_s, could be adapted to meet additional system constraints such as energy conservation, minimum spectral occupancy, or multi-user frequency division multiple access (FDMA) [104], I proceed by assuming fixed values of B since, in general, these values are determined by standards, hardware limitations, and/or spectrum regulations. The focus is obtaining the best constellation $\mathcal{S} \in \mathbb{S}$ that maximizes the rate, while maintaining $\text{FER}(\cdot) \leq \mathsf{F}$. Equivalently I determine the SNR threshold for constellation $\mathcal{S}$, $\gamma_{\mathcal{S}}$, which is computed

$$P_B(\gamma_{\mathcal{S}}) = 1 - (1 - \mathsf{F})^{1/N_f}. \tag{2.4}$$

For example, consider link adaptation for a system with available constellation set $\mathbb{S} = \{\text{QPSK}, \text{16-QAM}, \text{64-QAM}\}$, $\mathsf{F} = 0.1$ (10% FER), and $N_f = 1000$ bits. Using (2.4) I find $P_B(\gamma_{\mathcal{S}}) \approx 1 \times 10^{-4}$ for each $\mathcal{S} \in \mathbb{S}$. Fig. 2.4 shows the BER and the SNR thresholds associated with each constellation. Link adaptation in this example chooses the highest throughput constellation, with throughput defined as $(1 - \text{FER}(\gamma, \mathcal{S}))K_{\mathcal{S}}B$, for digital constellation $\mathcal{S}$ at SNR γ. Fig. 2.5 shows the normalized throughput for fixed constellations and with link adaptation. Hence, link adaptation in narrowband single carrier systems is straightforward. A look up table resides in each wireless device to map SNR to the best constellation. The look up table is easily created through Monte

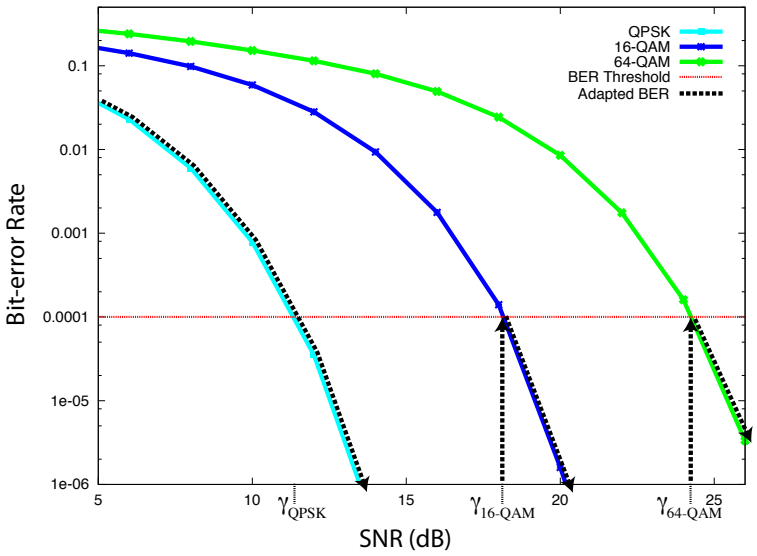


Figure 2.4: Bit-error rate vs. SNR for QPSK, 16-QAM, and 64-QAM. By adaptively selecting the constellation the reliability constraint is maintained for all SNR above γ_{QPSK}.

Carlo simulation of all constellations for all potential SNR realizations. It is also possible to use existing BER expressions for AWGN channels to determine SNR thresholds. For example, consider the tight BER upper bound for square QAM constellations of order M, also used in [33],

$$P_{B,\text{QAM}}(\gamma, M) \leq 0.2 \exp\left(\frac{-1.5\gamma}{M-1}\right) \tag{2.5}$$

which can be solved in terms of SNR to yield approximations of the SNR thresholds, i.e.,

$$\gamma_{M\text{-QAM}} \approx \frac{2}{3}(1-M)\log\left(5\mathsf{F}\right). \tag{2.6}$$

Unfortunately, these approximations may not always be accurate for all SNR or all values of M. Exact, more complex expressions are available for virtu-

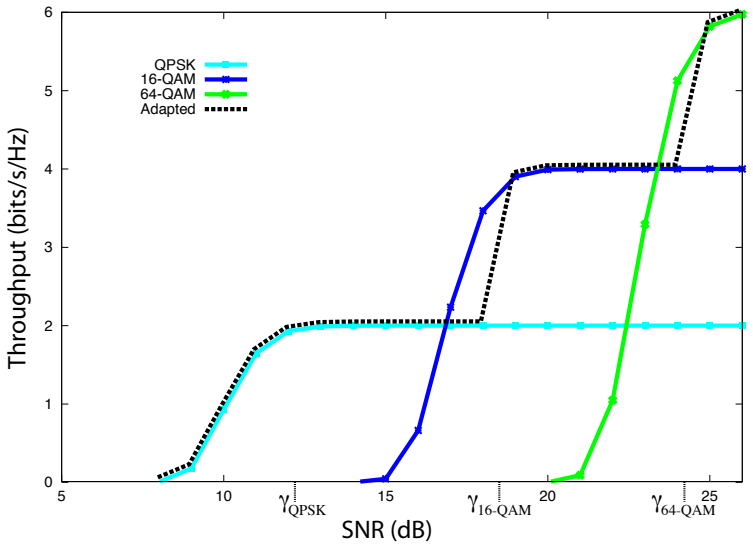


Figure 2.5: Throughput vs. SNR for QPSK, 16-QAM, and 64-QAM. Link adaptation allows us to maximize the throughput subject to reliability constraints. Note that link adaptation throughput slightly lags optimal throughput in order to address the reliability constraint.

ally all digital constellations in AWGN [54, 89, 97]. As an alternative to FER thresholds for reliability, a throughput optimal approach can also be considered by finding the throughput crossover points for each constellation in terms of SNR. Because this is a special case of the aforementioned FER constrained optimization with $\mathsf{F} = 1$, it is not treated explicitly in this dissertation.

2.2.3 Single Carrier Modulation in Frequency-Selective Channels

Modern wireless systems require large bandwidth for multimedia data traffic and consequently operate over broader spectrum. This results in selective fading over frequency at the receiver. Single carrier transmission of a single frame in frequency-selective channels is be modeled through the relationship

$$y[n] = \sqrt{E_s} \sum_{\ell=0}^{\nu} x[n-\ell]h[\ell] + v[n] \tag{2.7}$$

where the frequency-selective channel is modeled through the causal impulse response $\{h[\ell]\}_{\ell=0}^{\nu}$ with ν taps of delay spread. In narrowband systems the maximum likelihood symbol-by-symbol detection rule is straightforward to implement based on distance from constellation points scaled by $E_s|h|$. In frequency-selective channels, however, symbol-by-symbol detection is suboptimal due to intersymbol-interference (ISI), i.e., from taps $\{h[1], h[2], \ldots, h[\nu]\}$. If maximum likelihood sequence detection is used at the receiver, link adaptation is complicated because SNR no longer uniquely determines the BER. Most implementations, however, use equalizers to remove ISI effects and then proceed with symbol-by-symbol detection as in the frequency-flat channel.

Most equalizers can express the equivalent post-processing SNR. Although, for the most part, this results in colored noise, the post-processing SNR can be used to complete adaptation in the same manner as with frequency-flat channels. For example, zero-forcing (ZF) and minimum mean square error (MMSE) are two common linear equalizers with post-processing SNR equal to

$$\gamma_{\text{ZF}} = \frac{E_s}{\sigma^2} \times \left(\int_{-\frac{1}{2}}^{\frac{1}{2}} \left(\sum_{\ell=0}^{\nu} e^{-j\ell 2\pi f} h[\ell] \right)^{-1} df \right)^{-1} \tag{2.8}$$

and

$$\gamma_{\text{MMSE}} = \frac{E_s}{\sigma^2} \times \left(\int_{-\frac{1}{2}}^{\frac{1}{2}} \left(\sum_{\ell=0}^{\nu} e^{-j\ell 2\pi f} h[\ell] + \frac{\sigma^2}{E_s} \right)^{-1} df \right)^{-1}, \tag{2.9}$$

respectively. More complicated equalizers have also been developed such as the decision feedback equalizers (DFE), which leverages information from past estimates to improve performance [14].

2.2.4 Orthogonal Frequency Division Multiplexing

In this dissertation I will not be concerned with single carrier transmission in frequency-selective channels, since OFDM allows me to avoid the deleterious effects of ISI. The fundamental details of OFDM are outside the scope of this dissertation. An excellent reference is provided in [35]. Consider OFDM formatted with N total subcarriers. Therefore, the channel for subcarrier k is expressed through the discrete Fourier transform (DFT)

$$\mathsf{H}[k] = \sqrt{\frac{1}{N}} \sum_{\ell=0}^{\nu} h[\ell] \exp\left(-j2\pi k\ell/N\right) \tag{2.10}$$

where $\nu \leq \min\{N_{cp}+1, N+1\}$ for cyclic prefix with N_{cp} symbols. If these conditions hold, symbol-by-symbol ML detection is optimal in the frequency domain (after cyclic prefix removal). Therefore, the complete set of SNR values for all subcarriers is $\{\gamma[k]\}_{k=0}^{N-1}$ where

$$\gamma[k] = \frac{E_s|\mathsf{H}[k]|}{\sigma^2} \tag{2.11}$$

for all $k \in \{0, 1, \ldots, N-1\}$. Clearly, the variability of the SNR over the subcarriers depends on the statistical distribution for $\{h[\ell]\}_{\ell=0}^{\nu}$.

Although symbol-by-symbol detection is optimal in the frequency domain, reliability analysis is still very complicated because FER, for example, is a function of an N-dimensional link quality metric. Hence, link adaptation is not straightforward because it is not-feasible to create a N-dimensional look up table. Fortunately, there has been significant research on reducing the dimensionality of link quality metrics down to one. I defer this discussion until after Section 2.2.7 since all practical OFDM wireless systems also include error control coding to maximize spectral efficiency and protect against frequency selective fades.

Because the channel quality varies amongst OFDM subcarriers, it is advantageous to vary the transmit power dedicated to each subcarrier. For example, to optimize the capacity over all subcarriers I proceed with the classical waterfilling procedure, solved by maximizing the Lagrangian

$$\frac{1}{2}\sum_{k=0}^{N-1}\log_2\left(1+\frac{E_s[k]|\mathsf{H}[k]|}{\sigma^2}\right)+\lambda\left(\sum_{k=0}^{N-1}E_s[k]-E_s\right) \tag{2.12}$$

in terms of $E_s[k]$ with Lagrangian constraint parameter λ which leads to a constant value for $E_s + \sigma^2/|\mathsf{H}[k]\ \forall k$. Hence, optimization is iteratively solved by eliminating subcarriers in each iteration that require negative energy. The premise behind waterfilling is that the number of bits allocated to each subcarrier will vary. In practical systems this means that the digital symbol constellation can vary per subcarrier. Although this practice has been very successful for digital subscriber lines, where the channel is static for a very long time, such is not the case in wireless links. Consequently, bit-loading is not common practice in wireless networks because it requires more overhead than rate gain versus link adaptation where the same digital symbol constellation is chosen over each subcarrier, especially if there is not enough training to obtain accurate channel estimation at the receiver [4]. This is particularly true when forward error correction is considered. For fixed rate systems, the waterfilling procedure can be leveraged to maximize the energy margin. For a more complete discussion of power and bit-loading for OFDM the reader is encouraged to consult [14].

2.2.5 Multiple-Input, Multiple-Output Systems

Transmitting and receiving with multiple antennas in the multiple-input, multiple-output system model (MIMO) adds new dimensions to the wireless channel. In general, two distinct MIMO strategies exploit multiple antennas: diversity and multiplexing. Diversity strategies add redundancy over spatial dimensions, resulting in SNR gain (which can then be translated

to a rate gain through link adaptation). Popular diversity strategies include space-time codes, space-frequency codes, beamforming, and antenna selection. Multiplexing strategies send independent data over spatial dimensions, providing a direct rate gain. The number of independent data streams or *spatial streams* is defined N_s and is fundamentally limited by the number of transmit and receive antennas, i.e., $N_s \leq \min\{N_r, N_t\}$. Theoretical studies have expressed a unique tradeoff between diversity and multiplexing that characterizes a channel realization [107], although its utility in practice is yet to be defined.

A discrete-time impulse response is used to represent the frequency-selective channel between each transmit and receive antenna. The discrete-time impulse response between transmit antenna $i_t \in \{1, 2, \ldots, N_t\}$ and receive antenna $i_r \in \{1, 2, \ldots, N_r\}$ is denoted $\{h_{i_r,i_t}[\ell]\}_{\ell=0}^{\nu}$. With MIMO-OFDM I create the matrix $\mathbf{H}[k]$ for subcarrier k such that

$$[\mathbf{H}[k]]_{i_r,i_t} \triangleq \sqrt{\frac{1}{N}} \sum_{\ell=0}^{\nu} h_{i_r,i_t}[\ell] \exp\left(-j2\pi k\ell/N\right) \tag{2.13}$$

for $\mathbf{H}[k] \in \mathbb{C}^{N_r \times N_t}$. Next I examine the operation of space-time and space-frequency coding for diversity in MIMO. Afterwards I discuss the joint operation of spatial multiplexing and linear precoding, where linear precoding is the generalization of beamforming to multiple streams.

Space-Time/Frequency Coding: A comprehensive study of space time or space frequency coding is outside the scope of this dissertation. Here I extend a simple example of Alamouti space-time and space frequency coding

with 2 transmit antennas and show how the net SNR gain can be predicted. This predicted gain can then be used by link adaptation algorithms to determine adaptation thresholds.

Consider a narrowband MIMO channel matrix $\mathbf{H} \in \mathbb{C}^{N_r \times 2}$ with thermal noise vector $\mathbf{v} \in \mathbb{C}^{N_r}$ where each element is i.i.d. distributed $\mathcal{NC}(0, \sigma^2)$. Space-time coding provides diversity in the transmission of 2 data symbols $x_1, x_2 \in \mathbb{C}$ over two time slots. Simple repetition coding over alternate transmit antennas provides independent looks at the noise and diversity in the wireless channel, but repetition does not have nice decoding properties. Alamouti devised a space-time coding procedure where in the first time slot x_1 is transmitted on the first antenna and $-x_2^*$ is transmitted on the second antenna. In the second time slot x_2 is transmitted on the first transmit antenna and x_1^* is transmitted on the second antenna. Let the received signals for both of these time slots be

$$\mathbf{y_1} = \mathbf{H} \begin{bmatrix} x_1 \\ -x_2^* \end{bmatrix} + \mathbf{v_1} \tag{2.14}$$

$$\mathbf{y_2} = \mathbf{H} \begin{bmatrix} x_2 \\ x_1^* \end{bmatrix} + \mathbf{v_2} \tag{2.15}$$

where $\mathbf{v_1}, \mathbf{v_2}$ is the thermal noise vector for both time slots. Let $\mathbf{G} \in \mathbb{C}^{N_r \times N_r}$ be the linear decoding matrix where

$$\mathbf{G} = \begin{bmatrix} [\mathbf{H}]^*_{:,1} & \mathsf{conj}\left([\mathbf{H}]^*_{:,2}\right) \\ \mathsf{conj}\left(-[\mathbf{H}]^*_{:,2}\right) & [\mathbf{H}]^*_{:,1} \end{bmatrix} \tag{2.16}$$

such that

$$\mathbf{G} \begin{bmatrix} \mathbf{y_1} & \mathsf{conj}\left(\mathbf{y_1}\right) \\ \mathsf{conj}\left(\mathbf{y_2}\right) & \mathbf{y_2} \end{bmatrix} = \begin{bmatrix} \tilde{y}_1 \\ \tilde{y}_2 \end{bmatrix} \tag{2.17}$$

yields the two equivalent parallel SISO outputs

$$\tilde{y}_1 = \mathsf{tr}\left(\mathbf{H}^*\mathbf{H}\right) x_1 + \tilde{v}_1 \tag{2.18}$$

$$\tilde{y}_2 = \mathsf{tr}\left(\mathbf{H}^*\mathbf{H}\right) x_2 + \tilde{v}_2 \tag{2.19}$$

with $\tilde{v}_1, \tilde{v}_2 \sim \mathcal{NC}(0, N_r\sigma^2)$. Hence, linear decoding provides straightforward maximum likelihood detection and the SNR after linear decoding is

$$\gamma_{\text{Alamouti}} = \frac{E_s \mathsf{tr}\left(\mathbf{H}^*\mathbf{H}\right)^2}{N_r\sigma^2}. \tag{2.20}$$

The same coding can be applied in MIMO-OFDM systems by processing Alamouti over different subcarriers, resulting in space-frequency coding. Care must be taken, however, to ensure that the subcarriers display the same channel matrices. Significant research has been dedicated towards extending Alamouti to more than 2 transmit antennas. Unfortunately, extending codes to higher dimensions results in a combination of lost rate (less than one symbol is transmitted per time slot), lost diversity, and/or increased decoding complexity. I will not proceed with further details of space-time/frequency coding except to say that link adaptation must evaluate the expected SNR gain and rate associated with the code to determine if it provides a throughput advantage and should be selected as a transmission parameter.

Spatial Multiplexing and Linear Precoding: I assume the use of a precoding matrix for each subcarrier $\mathbf{F}[k] \in \mathbb{C}^{N_t \times N_s}$ where N_s is the number of spatial streams used during transmission. Power control over spatial streams and subcarriers may be embedded in these precoding matrices. I will not

explicitly consider the selection algorithm to determine the precoding matrix, however it is implicitly treated by assuming the precoding matrix is part of the effective channel matrix $\mathbf{H}[k]\mathbf{F}[k]$. For example, open loop precoding with equal power distribution on different spatial streams would result in $\mathbf{F}[k] = \sqrt{\frac{E_s[k]}{N_s}}\mathbf{I}$. For a detailed discussion of precoding in wireless systems, please consult [60].

Detection in MIMO systems parallels detection in single-carrier frequency selective systems. Maximum likelihood detection depends on all symbols transmitted over all streams. To perform stream-by-stream detection, i.e. to decouple the streams, I use an equalizer. For example, consider the linear equalization matrix $\mathbf{G}[k]$ such that the equalized estimate of the input symbol vector becomes $\mathbf{G}[k]\mathbf{H}[k]\mathbf{F}[k] + \mathbf{G}[k]\mathbf{v}[k]$ for thermal noise vector $\mathbf{v}[k]$ on subcarrier k where each element of the vector $\sim \mathcal{NC}(0, \sigma^2)$. It is also possible to decouple streams at the transmitter if channel state information is available through SVD precoding. That is if $\mathbf{F}[k] = [\mathbf{\Phi}[k]]_{1:N_t,1:N_s}$ for $\mathbf{H}[k] = \mathbf{\Theta}[k]\mathbf{\Sigma}[k]\mathbf{\Phi}[k]^*$.

If the streams are decoupled through precoding and/or equalization, the set of post-processing SNR over all subcarriers *and* spatial streams completely determines the reliability of transmission. That is

$$\mathrm{FER}(\{\mathbf{H}[k]\}_{k=0}^{N-1}, \{\mathbf{F}[k]\}_{k=0}^{N-1}, \{\mathbf{G}[k]\}_{k=0}^{N-1}, \sigma^2) = \mathrm{FER}(\{\{\gamma[a,k]\}_{a=1}^{N_s}\}_{k=0}^{N-1}) \tag{2.21}$$

where the post-procesing SNR per spatial stream $a \in \{1, 2, \ldots, N_s\}$ as

$$\gamma[a,k] \triangleq \frac{E_s |\left[\mathbf{G}[k]\mathbf{H}[k]\mathbf{F}[k]\right]_{a,a}|^2}{\sum_{a' \neq a} E_s |\left[\mathbf{G}[k]\mathbf{H}[k]\mathbf{F}[k]\right]_{a,a'}|^2 + \sigma^2 \sum_{a'=1}^{N_s} |\left[\mathbf{G}[k]\right]_{a,a'}|^2} \tag{2.22}$$

in subcarrier k [42]. Common spatial equalizers include ZF where $\mathbf{G}_{\mathrm{ZF}}[k] \triangleq (\mathbf{F}^*[k]\mathbf{H}^*[k]\mathbf{H}[k]\mathbf{F}[k])^{-1}\mathbf{F}^*[k]\mathbf{H}^*[k]$ and MMSE with spatial equalization matrix $\mathbf{G}_{\mathrm{MMSE}}[k] \triangleq (\mathbf{F}^*[k]\mathbf{H}^*[k]\mathbf{H}[k]\mathbf{F}[k] + (\sigma^2/E_s)\mathbf{I})^{-1}\mathbf{F}^*[k]\mathbf{H}^*[k]$. For ZF receivers, post-processing SNR simplifies to $\gamma_{\mathrm{ZF}}[a,k] = E_s/(\sigma^2 \sum_{a'=1}^{N_s} |[\mathbf{G}_{\mathrm{ZF}}[n]]_{a,a'}|^2)$, with the desirable property of Gaussian inter-stream interference plus noise. This further complicates link adaptation because I now have to consider a factor N_s more dimensions in the link quality metric.

2.2.6 MIMO Switching Metrics

A MIMO switching metric quantitatively evaluates the tradeoff between diversity and multiplexing in practice. For example, consider a narrowband MIMO system with channel matrix $\mathbf{H} \in \mathbb{C}^{N_r \times N_t}$ where $N_r = N_t$. Its Demmel condition number is given by

$$\kappa_D = \|\mathbf{H}_F\| \|\mathbf{H}^{-1}\|_2. \tag{2.23}$$

[41] showed that the Demmel condition number exactly characterize the operating regions where Alamouti and orthogonal space-time codes are preferable to spatial multiplexing with $N_s = N_t = N_r$ spatial streams in uncoded narrowband transmission with maximum likelihood detection. In practice, Monte Carlo simulations can generate a look-up-table to determine the cross-over

points between space-time codes and multiplexing in terms of the Demmel condition number. Intuitively, this concept can also be generalized to enable a switching metric that allows $N_s < N_t = N_r$ spatial streams. The Demmel condition number has an alternative representation as the root sum of the squared singlular values divided by the minimum singular value. Hence, a switching criterion based on the ratio of the root sum of the 1st through N_s-largest squared singular values to the N_s-largest singular value provides a similar switching metric.

Adaptation complexity is reduced through switching metrics. Switching metrics prevent consideration of diversity or multiplexing transmission modes if they are not preferable. In MIMO-OFDM with frequency selective channels, however, accurate switching metrics can be difficult to produce since different subcarrier channel matrices may exhibit different diversity and multiplexing preferences.

2.2.7 Forward Error Correction

Error correcting codes applied at the transmitter allow wireless systems to protect against intermittent errors in receiver detection. This is especially valuable for OFDM and MIMO systems which often experience subcarriers/spatial streams with inferior SNR due to frequency- and spatial-selective fading, respectively. Error correcting codes also improve the spectral efficiency. Consider Figure 2.6, for example, which shows BER curves for 64-QAM for uncoded and multiple convolutional coding rates in a narrowband channel.

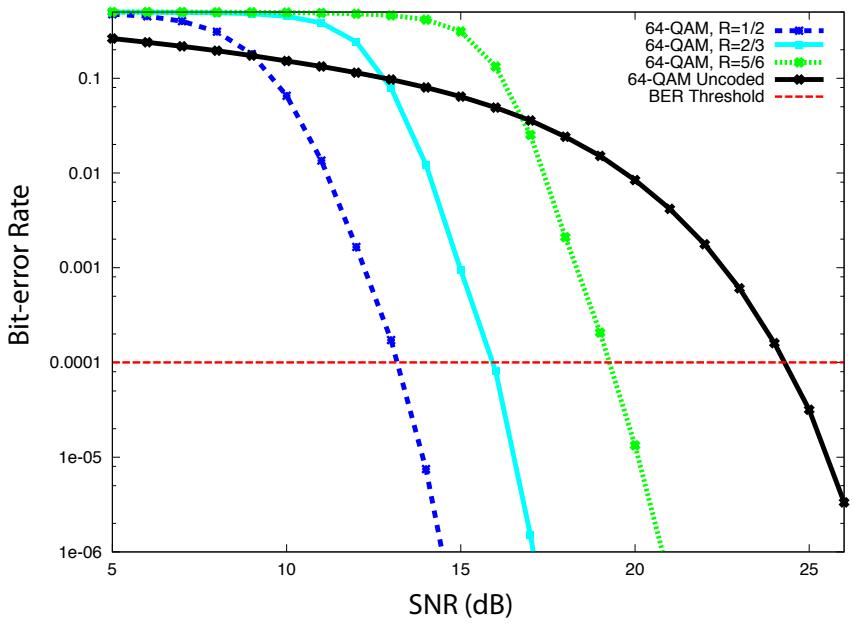


Figure 2.6: Bit-error rate vs. SNR for 64-QAM and convolutional coding with $\{133\}_8$ and $\{171\}_8$ generators and variable coding rates in narrowband channels.

By comparing Figures 2.4 and 2.6 it is observed that, even though uncoded 16-QAM and 64-QAM with rate 2/3 coding offer the same raw rate, the coded BER is superior. Hence, by modifying the digital symbol constellation and error control coding rate, link adaptation reduces the gap between capacity and practical data rates.

Unfortunately, forward error correction also leads to significant complexity in the performance analysis of frequency- and spatial-selective wireless channels. In OFDM, for example, the SNR per subcarrier is the result of partially correlated fading, where the proportions depend on the delay spread, the power delay profile, the number of subcarriers, and the subcarrier bandwidth

[103]. Similarly, MIMO systems result in variable SNR per spatial stream due to unequal singular values in the channel matrix. Further, interleaving over subcarriers and spatial streams increases the randomness of fades, which improves performance in these selective environments, but adds to analysis difficulties. A general and exact expression of FER is not known, especially for soft decoders where error events depend on the channel quality for all subcarriers/spatial streams involved in the error event. Since the number of potential error events is very large, extracting simple link quality metrics is not straightforward.

2.2.8 Link Quality Metrics

SNR alone is not a sufficient link quality metric in frequency and spatial selective channels. In [68] analysis suggested, however, that a single-dimensional quantity link quality metric (effective SNR) could be used to represent the post-processing SNR over all OFDM subcarriers in frequency-selective channels. This suggestion can also be extended to multiple spatial streams for MIMO-OFDM systems. This resulted in the discovery of many one–dimensional link quality metrics. A-single dimension link quality metric can be mapped to FER for each MCS using look-up-tables. A look-up-table is filled out based on measurements/simulations of FER in terms of the link quality metric. Look-up-tables, if populated through measurements, can account for model violations and do not rely on extensive system analysis. Most link adaptation research also provides mappings between link quality metrics

of different MCS to reduce storage complexity.

The following is a list of link quality metrics for (MIMO)-OFDM.

- average SNR $\rightarrow$ LQM $= \frac{1}{NN_s} \sum_{a=0}^{N_s-1} \sum_{k=0}^{N-1} \gamma[a,n]$
- uncoded BER $\rightarrow$ LQM $= \frac{1}{NN_s} \sum_{a=0}^{N_s-1} \sum_{k=0}^{N-1} Q\left(\sqrt{2\gamma[a,k]}\right)$
- minimum SNR $\rightarrow$ LQM $= \min_{a,k} \{\gamma[a,n]\}$
- minimum capacity $\rightarrow$ LQM $= \min_{a,k} \{\log_2(1+\gamma[a,k])\}$
- effective exponential SNR mapping (EESM) $\rightarrow$ LQM $= \frac{\beta}{NN_s} \log\left(\sum_{a=0}^{N_s-1} \sum_{k=0}^{N-1} \exp(-\gamma[a,k]/\beta)\right)$ where β is a scalar tuned for the selected MCS
- average capacity $\rightarrow$ LQM $= \frac{1}{NN_s} \sum_{a=0}^{N_s-1} \sum_{k=0}^{N-1} \log_2(1+\gamma[a,k])$
- mean mutual information $\rightarrow$ LQM $= \frac{1}{NN_s} \sum_{a=0}^{N_s-1} \sum_{k=0}^{N-1} \mathcal{I}(\gamma[a,k])$ where $\mathcal{I}(\cdot)$ is the mutual information for each symbol in a fixed MCS

The main problem with these link quality metrics is that they do not sufficiently capture the frequency and spatial selectivity of the channel. For example, the popular exponential effective SINR mapping (EESM) link quality metric is an average, over all subcarriers and spatial streams, of the exponentially mapped subcarrier SNR (β is a constant tuned to each MCS). EESM can capture coarse information about the channel state, which is sufficient for single-antenna channels with modest frequency selectivity. However, in spatial-selective and severe frequency-selective channels, a more precise characterization of the channel state is required. As a consequence, as demonstrated

in [48] and in Chapter 3 of this dissertation, EESM (as with all current link quality metrics) performs poorly. Hence, to address future wireless standards which will inevitably operate over broader bandwidths and with more spatial streams, new link quality metrics are desired for MIMO-OFDM. For a more specific discussion of link quality metrics, please consult the following references [8, 9, 12, 25, 48, 52, 53, 87, 94].

2.2.9 Effect of Model Violations

Practical transceivers exhibit nonlinearities in a multitude of hardware components including power and low-noise amplifiers, mixers, and analog filters [27, 99, 108]. Nonlinearities in the power amplifier are particularly well studied and many models exist to emulate their behavior. To demonstrate the impact nonlinear power amplifiers have on link adaptation, Figure 2.7 shows bit-error rate (BER) with power amplifier nonlinearity. Because this simulation depicts single-carrier narrowband communication, the signal variance to noise variance ratio (SNR) is the standard link quality metric. If the receiver assumes the linear, time-invariant, additive white Gaussian noise (LTI-AWGN) model in a system with amplifier nonlinearity, the reliability predicted by measuring SNR is skewed. Amplifiers can improve their linearity by backing off the transmitted power, but this has diminishing returns. Moreover, the receiver loses significantly more SNR during backoff than is gained (assuming noise limited environment). Clearly, it is preferable to operate under conditions of minimal backoff, but this is not possible because the link

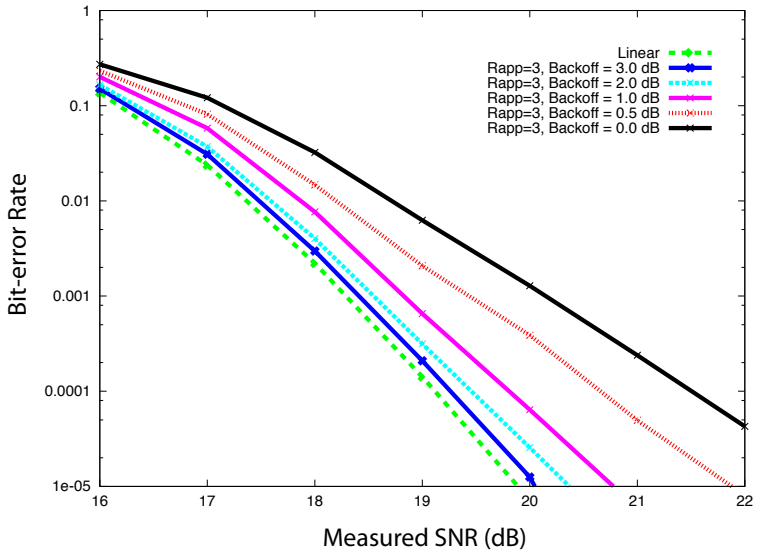


Figure 2.7: Bit-error rate vs. measured SNR of 64-QAM with rate 1/2 convolutional codes and Rapp nonlinear solid-state power amplifier mode (Rapp parameter = 3 corresponding to 1 Watt solid state amplifier) [99].

adaptation algorithm, which is based on the LTI-AWGN model, will choose incorrect communication parameters that either sacrifice rate, reliability, or both.

Non-white and non-Gaussian additive noise is also pervasive in wireless links. Non-ideal filters color thermal noise in the receiver. Further, finite resolution in digital-to-analog/analog-to-digital (DAC/ADC) converters, local oscillator instability (phase noise), and ambient interference all contribute noise with wildly varying statistical distributions [15, 23]. The heavy-tailed phase noise has increasingly become a problem in wireless systems due to the push towards higher operating frequencies [20, 81]. Figure 2.8, for example, shows the BER impact of changing the distribution order of generalized Gaus-

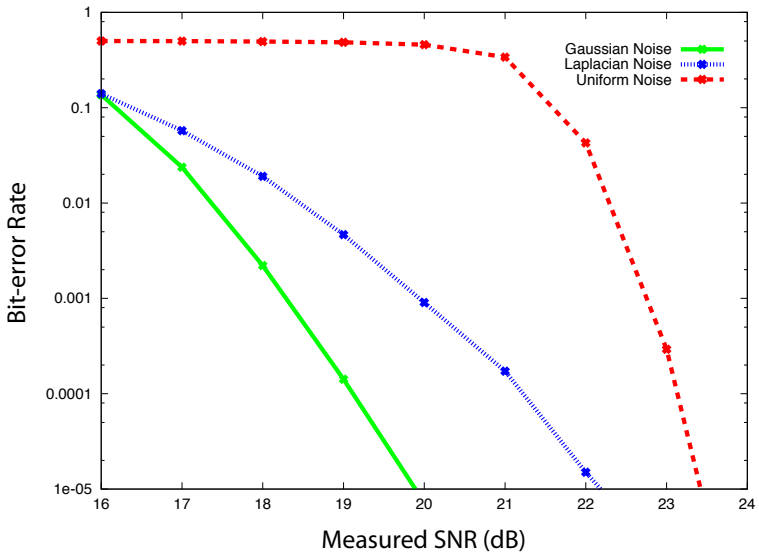


Figure 2.8: Bit-error rate vs. measured SNR (receiver assumes white Gaussian noise) for various realizations of additive noise. Some studies have shown that phase noise in OFDM may be represented by Laplacian noise [7]. For QAM constellations, a zero-mean uniform distribution approximates the noise distribution in interference-limited environments.

sian additive noise in a receiver designed for AWGN. Again, model violations skew reliability predictions by the link adaptation algorithms based on AWGN models. Clearly, if additive noise distributions deviate from AWGN, this can have drastic link adaptation consequences.

It should be noted that error vector measurement (EVM) requirements in standards intend to limit model violation effects. Nevertheless, EVM requirements cannot be too strict since low-cost hardware is of paramount importance. We can expect that, even with EVM requirements, permissible model violations in the hardware and ambient interference will still shift the expected

performance in terms of the link quality metrics. Further, it is unlikely that all fielded devices will strictly adhere to EVM requirements.

2.2.10 Link Adaptation in Standards

I now summarize the extent of link adaptation in current wireless standards that feature MIMO-OFDM. I will use this summary to develop the scope of link adaptation considered in this dissertation.

3GPP-LTE: The third generation partnership project, long-term evolution (3GPP-LTE) represents the global cellular-carrier-supported data access standard, often used synonymously with 4G for the 4th generation of wireless cellular access. I will focus on the salient features of E-UTRA release 8 [17].

Resource Allocation: LTE uses orthogonal frequency division multiple access (OFDMA) to multiplex users in the downlink (base station to user). OFDMA is combined with time division multiple access (TDMA) by assigning different OFDM subcarriers in different time slots to different users. In the uplink (user to base station) LTE applies single-carrier frequency division multiple access (SC-FDMA). In SC-FDMA, users generate single-carrier signals which are then mapped to time/frequency resources through a DFT/IDFT operation, making it similar conceptually to OFDMA. Although, theoretically, OFDMA is preferred due to nice equalization properties in broadband links, SC-FDMA is preferred over OFDMA in the practical uplink since it allows for a greater tolerance of imperfect hardware, for example through lower signal peakedness (i.e., a lower peak-to-average power ratio (PAPR)). In the down-

link, the base station is the only device delivering data, so users only have to synchronize with a single frequency offset, making OFDMA more desirable.

LTE also offers allocation of resources in the spatial domain for the downlink, which is enabled by the availability of multiple antennas at the base station and user. In general, time, frequency, and spatial resources are scheduled independently of the constellation and FEC configuration. This means that link adaptation (as it is considered in this dissertation) may only occur over a set of subcarriers and spatial streams assigned to a single user in a fixed resource (equal to an integer multiple of OFDM symbols) that observes the slow fading assumption. Hence the frequency and spatial selectivity of the channel that link adaptation must account for inherently depends on the resources a user is allocated as well as the power control algorithm implemented over these resources.

Subcarrier Symbol Constellation Formats: In the downlink, LTE supports QPSK, 16-QAM, and 64-QAM constellations. In the uplink, the same constellations are supported although 64-QAM is optional, depending on the capabilities offered by the user device.

MIMO Processing: LTE provides a full range of MIMO processing capabilities including space-frequency block coding, precoding, and spatial multiplexing support for up to four spatial streams, although space-frequency block coding can only be used on control and broadcast channels. Multi-user MIMO is also supported such that user resources can be split among spatial streams.

Forward Error Correction: LTE primarily uses turbo coding at a nominal 1/3 coding rate (tail biting convolutional codes are used on control channels). A circular buffer implementation allows for rate matching that is compatible with Hybrid-ARQ. This implementation allows for arbitrary coding rates, although overhead issues require a reduced set of rate and constellation pairs for user feedback $\{0.076, 0.12, 0.19, 0.3, 0.44, 0.59\}$ for QPSK, $\{0.37, 0.48, 0.6\}$ for 16-QAM, and $\{0.45, 0.55, 0.65, 0.75, 0.85, 0.93\}$ for 64-QAM.

Hybrid-ARQ: In standard ARQ failed communication of a frame over a wireless link requires a retransmission of all bits in the packaged frame. Hybrid-ARQ allows for incremental redundancy by forwarding additional parity bits from a buffered and encoded frame if a NACK for that frame is received. This reduces overhead associated with ARQ and also diminishes the risk of incorrectly adapting to the channel. LTE provides for asynchronous and adaptive Hybrid-ARQ where incremental redundancy does not have to occur at regular intervals and also is not required to use the same constellation and MIMO processing as the initial frame transmission.

Feedback: In the uplink, the base station is aware of the user channels through training reference signals. Therefore, the base station informs the user of the uplink transmission parameters in addition to the allocated resources. To enable downlink link adaptation and MIMO processing in LTE, the procedure is more complicated. Broadly, this is accomplished through feedback of the user desired transmission parameters for frequency resource subdivisions: wideband and subband. In the wideband scenario, the user recommends the

transmission parameters assuming the entire system bandwidth is allocated. In the subband case, the user recommends the unique transmission parameters for multiple separate blocks of contiguous subcarriers.

Link Adaptation Comments: LTE assumes that hybrid-ARQ and link adaptation operate independently. Therefore, the constellation, coding rate, and MIMO transmission mode are selected based on feedback/channel estimates in the downlink/uplink, respectively after power control and resource allocation have been completed. LTE supports single- and multi-codeword MIMO. In single-codeword MIMO, the same constellation and coding rate is selected over all subcarriers and spatial streams. In multi-codeword MIMO, the same constellation is selected over all subcarriers and spatial streams for each codeword. Up to two codewords are supported.

WiMax: WiMax offers an alternative mobile broadband data protocol that has developed a sizable market recently despite not enjoying global acceptance by cellular providers. The WiMax profile is based on the IEEE 802.16e standard (soon to be superseded by IEEE 802.16m) and its design is very similar to LTE. I will focus on the more mandatory features of Mobile WiMax Release 1.5 [2].

Resource Allocation: Like LTE, Mobile WiMax also uses OFDMA, but for both downlink and uplink with time division duplexing (TDD) or frequency division duplexing (FDD). Each user is allocated blocks of subcarriers or subchannels. Each subchannel may contain contiguous or distributed groups of subcarriers. In addition to subchannel resource allocation, users may be flex-

ibly multiplexed in time.

Subcarrier Symbol Constellation Formats: In the downlink, WiMax supports BPSK (optional), QPSK, 16-QAM, and 64-QAM constellations. In the uplink, the same constellations are supported although 64-QAM is optional, depending on the capabilities offered by the user device.

MIMO Processing: WiMax includes support for space-time block coding, precoding, and spatial multiplexing support for up to two spatial streams. Multi-user MIMO is also supported in the uplink such that user resources can be split among two spatial streams.

Forward Error Correction: WiMax requires convolutional coding with rate 1/2, 2/3, 3/4, and 5/6. Interleaving is processed over subcarriers and spatial streams.

Hybrid-ARQ: As with LTE, WiMax provides for asynchronous and Hybrid-ARQ with incremental redundancy.

Feedback: In the uplink, the base station is aware of the user channels through training reference signals. Therefore, the base station informs the user of the uplink transmission parameters in addition to the allocated resources. To enable downlink link adaptation, coarse measurements of the channel quality based on signal to interference plus noise ratios (SINR) and signal strength indicators (RSSI) are fed back to the base station.

Link Adaptation Comments: Link adaptation in WiMax is similar to LTE, with the exception of the differences in feedback and the absence of

multi-codeword MIMO.

IEEE 802.11n: IEEE 802.11n is the latest international standard for WLAN [43].

Resource Allocation: IEEE 802.11n uses random a access MAC and users are multiplexed in the time domain. For a given time slot, which is provided through a contention mechanism with random backoff, each user is allocated 56 data and pilot subcarriers with 8 guard subcarriers through OFDM for a 20 MHz channel spectrum. For a 40 MHz channel, each user is allocated 114 data and pilot subcarriers with 14 guard subcarriers.

Subcarrier Symbol Constellation Formats: IEEE 802.11n offers BPSK, QPSK, 16-QAM, and 64-QAM constellations.

MIMO Processing: IEEE 802.11n supports space-time coding (optionally), spatial multiplexing (up to 4 spatial streams (2 mandatory), and precoding.

Forward Error Correction: IEEE 802.11n requires convolutional coding with rate 1/2, 2/3, 3/4, and 5/6. IEEE 802.11n optionally features LDPC codes. Interleaving is processed over subcarriers and spatial streams.

Hybrid-ARQ: IEEE 802.11n does not support hybrid-ARQ.

Feedback: IEEE 802.11n supports feedback of the constellation and coding rate, the wireless channel impulse response, and the beamforming or precoding matrices.

Link Adaptation Comments: Link adaptation in IEEE 802.11n is accomplished through selection of the precoding matrices, the space-time block code, the constellation, and the coding rate for each transmission. MIMO is only supported through a single codeword. Optional transmission modes allow for the constellation to vary per spatial stream in single-codeword MIMO.

2.2.11 Scope of Link Adaptation in this Dissertation

Although link adaptation can encompass all of the options mentioned in this chapter, I will focus on a limited set, namely the selection of digital symbol constellation, the forward error correction encoder configuration, and the number of spatial streams. Collectively I will refer to this as the modulation and coding scheme (MCS). As a consequence throughout this dissertation link adaptation and adaptive modulation and coding (AMC) will be used interchangeably. I will assume that all subcarriers use the same digital symbol constellation, because the overhead to enable bit-loading has not been justified for wireless systems with dynamic channels. Further, power allocation on subcarriers will be assumed as a separate process and will implicitly be observed as part of the effective channel. Precoding over multiple antennas is not treated as part of link adaptation in this dissertation since I assume that this can also be treated separate from AMC. Again, this becomes part of the effective channel. Finally, space-time/frequency coding will not be evaluated since I assume the diversity gains can be captured through the post-processing SNR. Hence, I assume that AMC design principles I develop without space-

time coding easily translate to the scenario where space-time codes are offered. In general, while joint link adaptation over all potential communication parameters is optimal, I assume that the performance gain of a joint approach is not prudent until AMC is optimized separately.

AMC in the downlink of 3GPP-LTE and WiMax is limited by the link quality metrics that are fed back to the base station. One of the primary motivations for this dissertation is to improve the performance of MIMO-OFDM link adaptation in frequency and spatial selective channels, such as experienced in LTE, WiMax, and IEEE 802.11n. If I constrain myself to the link quality metric feedback provided by the standards, I will not be able to design novel spatial and frequency selective link quality metrics that improve the performance of AMC for future standards. Because of this, I have chosen to conform to the IEEE 802.11n system model since it provides the most flexibility in AMC for MIMO-OFDM and the link quality metrics used for link adaptation.

2.3 MIMO-OFDM System Model

This section defines the MIMO-OFDM system model, summarized in Fig. 2.9, which represents the communication procedure of wireless signals in a practical MIMO-OFDM wireless link such as IEEE 802.11n. Small modifications to this system model can be made to align with 3GPP-LTE and WiMax standards. This is discussed wherever appropriate. The notation used throughout the system and notable stages of the transceiver are shown in

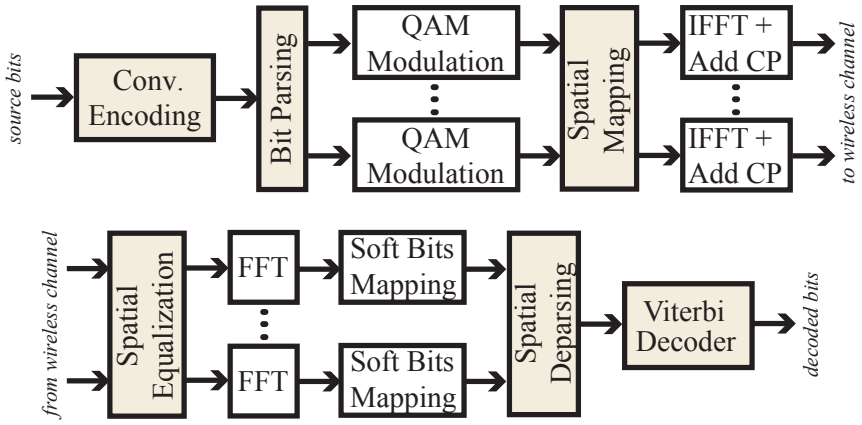


Figure 2.9: Transmitter and receiver processing for MIMO-OFDM system with bit-interleaved convolutional forward error correction

Tables 2.3 and 2.4, respectively. Each physical layer frame of the MIMO-OFDM transceiver is processed as follows.[4] The N_f source bits for a single frame $\{b[n_b]\}_{n_b=0}^{N_f-1}$ are delivered to the convolutional encoder in stage 1. In

[4]I assume positive integer values of $N_f/\mathcal{C}$ and $N_f/(\mathcal{C}KN_s)$. In practice this is accomplished through bit padding of the input frame. I also assume the convolutional encoder returns to its initial state through appropriate tail bits.

Table 2.3: Notation of MIMO-OFDM transceiver

	Description
N_f	frame length in bits (same as $\mathcal{C}KN_{\mathrm{O}}NN_s$)
$\mathcal{C}$	convolutional coding rate
K	number of bits per QAM symbol
N_O	number of OFDM symbols in frame
N	number of data subcarriers per OFDM symbol
N_t	number of RF chains in the transmitter
N_r	number of RF chains in the receiver
N_s	number of spatial streams ($\leq \min\{N_t, N_r\}$)

Table 2.4: Notable stage elements of MIMO-OFDM transceiver

Stage	**Description**	**Data Quantity**
1	source bits	$b[n_b] \in \{0,1\}$, $n_b \in \{0,1,\ldots,N_f-1\}$
2	encoded source bits	$c[n_c] \in \{0,1\}$, $n_c \in \{0,1,\ldots,N_f/\mathcal{C}-1\}$
3	baseband, frequency-domain transmitted QAM symbols	$\mathbf{X}[m,n] \in \mathbb{C}^{N_s}$ $m \in \{0,1,\ldots,N_\mathrm{O}-1\}$, $n \in \{0,1,\ldots,N-1\}$
4	equalized baseband frequency-domain received symbols	$\mathbf{Y}[m,n] \in \mathbb{C}^{N_s}$ $m \in \{0,1,\ldots,N_\mathrm{O}-1\}$, $n \in \{0,1,\ldots,N-1\}$
5	estimated source bits	$\hat{b}[n_b] \in \{0,1\}$, $n_b \in \{0,1,\ldots,N_f-1\}$

stage 2, the convolutionally-encoded bits $\{c[n_c]\}_{n_c=0}^{N_f/\mathcal{C}-1}$ with coding rate $\mathcal{C}$ are sent to the spatial bit parser. Each spatial branch $a \in \{1,2,\ldots,N_s\}$ of the transmitter receives an equivalent number of bits (I assume each spatial branch has the same modulation order and coding rate). The spatial parsing operation not only assigns bits from codewords to multiple spatial streams, but also interleaves bits over the spatial streams and subcarriers. In practical systems this function may not process the entire $N_f/\mathcal{C}$-length bit sequence at once due to complexity, latency, and memory constraints. In this paper I consider spatial parsing functions which process L bits consecutively.

After the spatial parsing operation, the bits in each spatial branch $a \in \{1,2,\ldots,N_s\}$ enter the M-ordered quadrature-amplitude modulator (M-QAM). Therefore, $K = \log_2(M)$ bits are modulated at a time in each spatial branch. Using matrix notation I represent the set of M-QAM samples across

all spatial branches as $\{\{\mathbf{X}[m,n]\}_{m=0}^{N_{\mathrm{O}}-1}\}_{n=0}^{N-1}$ in stage 3 where $\mathbf{X}[m,n] \in \mathbb{C}^{N_s}$. The spatial mapping block transforms each N_s-dimensional complex vector into N_t complex dimensions that map distinctly to each transmit radio frequency (RF) chain ($N_s \leq \min\{N_r, N_t\}$) through a linear precoding matrix $\mathbf{F}[n] \in \mathbb{C}^{N_t \times N_s}$ for each subcarrier n, although the exact linear precoding matrix used in this paper is not specified. The last digital processing at the transmitter is the inverse discrete Fourier transform (IDFT) and cyclic prefix (CP) addition. In each of the N_t transmit branches, the IDFT algorithm processes N-length blocks.[5]

At the receiver, the equalized samples are mapped equivalently into the discrete Fourier domain and at stage 4 the frequency-domain representation of OFDM symbol m and subcarrier n is given by $\mathbf{Y}[m,n]$.[6] After mapping the QAM symbols to soft bits, the receiver calculates the sequence of log-likelihood ratios. The optimal maximum-likelihood (ML) detection of MIMO signals, in general, does not use linear spatial equalization since soft bits are created from each N_r-dimensional received sample. If ideal precoding through the singular value decomposition (SVD) is implemented in the linear precoding matrix, however, soft bits may be created on each spatial stream separately [74]. The Viterbi decoder yields the estimated frame $\{\hat{b}[n_b]\}_{n_b=0}^{N_f-1}$ in stage 5.

[5]In practice not all N subcarriers carry data. Some subcarriers are dedicated to pilots or guardbands. In multiuser standards like 3GPP-LTE and WiMax, I only consider the subcarriers that are dedicated to a single user.

[6]Spatial equalization and the IFFT are often combined into a single step with the IFFT preceding spatial equalization since equalization in the frequency domain offers favorable complexity for broadband frequency selective wireless channels [26]. Fig. 2.9 provides a more general framework for equalization.

Given MIMO-OFDM symbol $m \in \{0, 1, \ldots, N_{\mathrm{O}} - 1\}$, subcarrier $n \in \{0, 1, \ldots, N - 1\}$, $\mathbf{H}[n] \in \mathbb{C}^{N_r \times N_t}$ as the wireless channel, $\mathbf{G}[n] \in \mathbb{C}^{N_s \times N_r}$ as the linear equalization matrix, and $\mathbf{V}[m, n] \in \mathbb{C}^{N_r}$ as the thermal noise at baseband, then

$$\mathbf{Y}[m, n] = \sqrt{E_s}\mathbf{G}[n]\mathbf{H}[n]\mathbf{F}[n]\mathbf{X}[m, n] + \mathbf{G}[n]\mathbf{V}[m, n]. \tag{2.24}$$

$\tilde{\mathbf{H}}[n] = \mathbf{G}[n]\mathbf{H}[n]\mathbf{F}[n]$ and $\tilde{\mathbf{V}}[m, n] = \mathbf{G}[n]\mathbf{V}[m, n]$ represent effective quantities to include the combined effects of spatial mapping and spatial equalization. All modulated signals will have unit average power such that $\mathbb{E}[\mathbf{X}^*[m, n]\mathbf{X}[m, n]] = 1 \; \forall \; m, n$. Consequently, E_s designates the expected total transmitted signal power if $\|\mathbf{F}[n]\|_F^2 = 1$. I will consider complex Gaussian noise vectors $\mathbf{V}[m, n] \sim \mathcal{CN}(\mathbf{0}, \sigma^2\mathbf{I})$ where each element has variance σ^2 and mean $0 \Rightarrow \tilde{\mathbf{V}}[m, n] \sim \mathcal{CN}(\mathbf{0}, \sigma^2\mathbf{G}[n]\mathbf{G}^*[n])$. The wireless channel $\mathbf{H}[n]$ for subcarrier n observes the slow fading assumption, provided the channel does not change for all OFDM symbols within a frame. This enables the receiver to remove m as a parameter of the effective channel.

The key assumptions for this system model are as follows.

1. *Uniform Modulation*: Every symbol over each spatial stream and every subcarrier is modulated with the same constellation order for each transmitted frame. This corresponds to the mandatory modes supported in IEEE 802.11n, WiMax, and single codeword MIMO in 3GPP-LTE. Multiple codeword MIMO, with parallel encoding, interleaving, and modulation operations, is available in 3GPP-LTE as well. Each of the codewords

are independent and therefore multiple codeword MIMO may be represented as multiple independent realizations of our current system model. In the optional modes of IEEE 802.11n different modulation orders may be considered for separate spatial streams. This is not covered in the current system model and is treated as future work.

2. *Linear Precoding*: All spatial mapping and precoding operations at the transmitter are accomplished through matrix operations (i.e. linear operations), although no assumption is made on the type of linear precoding used (open loop, channel informed, etc.). Nonlinear precoding techniques have been studied in the past [18], however they are not provided in current MIMO-OFDM standards. Moreover, nonlinear precoding methods complicate calculation of post-processing SNR, which is integral to the analysis in this paper.

3. *Linear Equalization*: Linear receivers reduce the complexity of decoding, especially for large dimension MIMO systems, by decoupling maximum likelihood sequence detection of convolutionally-coded bits and MIMO channel equalization. For systems that use SVD precoding at the transmitter, linear receivers achieve ML performance [74]. When SVD precoding is not used, however, linear receivers suffer performance degradation when compared to optimal ML over spatial streams. For simplicity I only consider linear receivers with the understanding that all development in this paper applies to alternative equalization procedures if SNR can be defined per subcarrier and spatial stream.

4. *Slow Fading*: The wireless channel is assumed static for all OFDM symbols in a single frame. This assumption is justified if the coherence time of the wireless channel is larger than the duration of the frame transmission. Slow fading is valid for indoor channels and many outdoor channels [77]. I do not consider the correlation of fading between two successively transmitted frames, only within each frame.

2.4 Machine Learning

In this section I summarize the relevant scope of machine learning in this dissertation. Due to length constraints, the reader is advised to consult any of the several references on machine learning for further details [6, 39, 67]. This section is not intended to be comprehensive, but instead to provide any background information that is essential for understanding link adaptation procedures developed in this dissertation.

A machine learning algorithm generates its response by evaluating patterns in example data. There are two main classes of machine learning algorithms: supervised and unsupervised. Supervised learning algorithms are given data sets along with the appropriate algorithm response to each element of the data set. In the unsupervised case, data sets are given without the appropriate system response.

2.4.1 Supervised Learning

Supervised learning algorithms evolve responses for two purposes: regression and classification.

Regression: Regression seeks a mapping from p-dimensional parameters to a real valued output. Formally, assuming a set of W known mappings from the parameter or *observation* space to the output range, collectively known as *training data*,

$$\{(\mathbf{z}_0, y_0), (\mathbf{z}_1, y_1), \ldots, (\mathbf{z}_{W-1}, y_{W-1})\} \tag{2.25}$$

with $\mathbf{z}_w \in \mathbb{R}^d$, $y_w \in \mathbb{R}$, supervised learning algorithms determine future mappings in the output range

$$f(\mathbf{q}) \in \mathbb{R} \tag{2.26}$$

for observation queries $\mathbf{q} \in \mathbb{R}^d$. Essentially, supervised learning algorithms determine $f(\cdot)$ which is empirically inferred from the training data.

Classification: More commonly, supervised learning provides classifiers. The training set for classification contains W known mappings from the observation space to class labels

$$\{(\mathbf{z}_0, y_0), (\mathbf{z}_1, y_1), \ldots, (\mathbf{z}_{W-1}, y_{W-1})\} \tag{2.27}$$

where $\mathbf{z}_w \in \mathbb{R}^p$, $y_w \in \mathcal{J} \subset \mathbb{N}$. Thus, for classification, supervised learning algorithms infer the class, indexed by a non-negative integer, that each new observation query $\mathbf{q} \in \mathbb{R}^p$ belongs to. In this dissertation I will consider the

application of classification algorithms to link adaptation where the classes equivalently map to a distinct set of transmission parameters.

Perhaps the most critical aspect of supervised learning is the preprocessing of data in the observation space. For link adaptation in multiple-antenna LTI-AWGN channels, raw data observations are merely the complex coefficients of the wireless channel impulse response for each transmit/receive antenna pair and the variance of the additive noise at each receive antenna. Hence, this results in $p = 2N_rN_t(\nu + 1) + N_r$. Supervised learning algorithms prefer that sufficient preprocessing is performed to reduce the dimensionality of the observation space down to small number of independent features that uniquely define the boundaries between classes. This process is known as *feature extraction*. In link adaptation, since the classes represent distinct transmission parameter realizations, the process of feature extraction is the same as the process of extracting a link quality metric. As a consequence, the terms link quality metric and *feature set* will often be used interchangeably throughout this dissertation. Note that look-up-table link adaptation also falls under the umbrella of supervised-learning based classification. Given the wealth of research on supervised learning and classification, this is an encouraging observation that suggests better results are almost certainly obtainable.

2.4.2 Unsupervised Learning

Unsupervised learning algorithms empirically infer information about the structure of data. Because the definition of unsupervised learning depends

on the algorithm implemented, I provide an example unsupervised learning algorithm used in this paper: principal component analysis (PCA).

Principal Component Analysis: Consider a training set of W data observations

$$\{\mathbf{z}_0, \mathbf{z}_1, \ldots, \mathbf{z}_{W-1}\} \tag{2.28}$$

which is assembled into the observation matrix

$$\mathbf{Z} = [\mathbf{z}_0 \mathbf{z}_1 \cdots \mathbf{z}_{W-1}] \tag{2.29}$$

for $\mathbf{Z} \in \mathbb{R}^{p \times W}$. PCA desires to find the most important p-dimensional components of this training set. Mathematically, PCA attempts to find an orthonormal basis matrix $\mathbf{B} \in \mathbb{R}^{p \times d}$ where the columns of the matrix form orthonormal basis vectors that may be combined to approximate any single observation in the training set. PCA accomplishes this through evaluation of the training data covariance matrix. Let $\mathbf{C}_{\mathbf{Z}} \in \mathbb{R}^{p \times p}$ denote the covariance matrix where

$$[\mathbf{C}]_{k,l} = \mathbb{E}\left[\left([\mathbf{z}]_k - \mathbb{E}\left[[\mathbf{z}]_k\right]\right)\left([\mathbf{z}]_l - \mathbb{E}\left[[\mathbf{z}]_l\right]\right)\right] \tag{2.30}$$

and $[\mathbf{z}]_k$ represents the kth element of any data observation vector for $k \in \{0, 1, \ldots, p-1\}$. Now consider the projection of the original data matrix for a change of basis through $\mathbf{B}$, i.e. $\mathbf{B}^*\mathbf{Z}$ with its associated covariance matrix

$$\mathbf{C}_{\mathbf{B}^*\mathbf{Z}} = \mathbf{B}^* \mathbf{C}_{\mathbf{Z}} \mathbf{B}. \tag{2.31}$$

If the basis vectors represent independent components of the training set, then $\mathbf{C}_{\mathbf{B}^*\mathbf{Z}}$ is diagonal and (2.31) requires that the columns of $\mathbf{B}$ are equal to the

eigenvectors of $\mathbf{C_Z}$. Moreover, the variance of each eigenvecor as a basis for the training set relates to the strength of the eigenvalue. Hence $\mathbf{B}$ is composed of the d eigenvectors in $\mathbf{C_Z}$ that correspond to the d largest eigenvalues. In this dissertation I use PCA to reduced dimensionality in link quality metrics for MIMO-OFDM channels.

2.5 Dataset Resources

To enable supervised learning for link adaptation, a publicly-available data set (on the WSIL web site [19]) of 2x2 MIMO-OFDM wireless system simulations using MCS_0-MCS_{15} of IEEE 802.11n have been completed. All packets are 128 bytes in length. The receiver uses soft-output zero-forcing spatial equalization combined with single-stream maximum likelihood decoding. Frequency offset and synchronization errors have not been considered. Space-time block codes are also not considered here. The system is simulated with 32000 channel realizations, using channel data set 1 and channel data set 2. Each channel is a realization where each matrix entry of each tap is a zero-mean independently and identically distributed complex-Gaussian random variable. Therefore this channel observes a uniform power delay profile. Different channel realizations have been chosen to represent a diverse selection of MCS within the channel set.

The channels completely characterize the input/output relationship of the system since the transmit power and noise variance are both normalized to one. The expected transmit power on each transmit antenna is 1, although

the instantaneous symbol energy is not normalized due to OFDM and QAM modulation. The receiver is modeled with additive complex Gaussian noise process where the real and imaginary component are independent and identically distributed normal random variables with mean 0 and variance $1/\sqrt{2}$. The channels and associated FER vectors are in IT++ data format. Therefore, they can be easily imported into C++ simulations using the IT++ library or into Matlab/Octave simulations using the itload.m function provided with the IT++ library. Each channel set is formatted as an 32000 dimensional array of complex matrices. Each of the 32000 complex matrices have N_r rows and νN_t columns where $N_r = N_t = 2$ and $\nu + 1$ is the total number of time-domain channel taps. Hence, the first N_t columns correspond to the first tap channel matrix. The N_t+1 through $2N_t$ columns correspond to the second tap channel matrix and so on. The FER vectors are 32000 dimensional where each entry corresponds to the FER simulation for a fixed MCS in one of the enumerated 32000 channels.

For 20 MHz channels (as modeled in these simulations), the FFT order of OFDM in IEEE 802.11n is 64. Therefore, the complex coefficient for the same matrix element in each of the 64 taps is concatenated and the FFT is processed to yield the channel matrix element for all 64 subcarriers (indexed 0 to 63). Note that none of the channels provided exceed 16 taps, so the remaining channel taps are just from zero padding. IEEE 802.11n does not use all subcarriers due to guard and null tones. Subcarriers indexed by $\{1 \rightarrow 6, 8 \rightarrow 20, 22 \rightarrow 28, 36 \rightarrow 42, 44 \rightarrow 56, 58 \rightarrow 63\}$ hold data, while subcarriers

indexed by $\{7, 21, 43, 57\}$ hold pilots. In general, only the channels for the data subcarriers are necessary to create feature sets for a channel realization since synchronization is assumed perfect in these simulations.

Chapter 3

Offline Supervised Learning for Link Adaptation

3.1 Introduction

As a first step to evaluate the potential of machine learning for link adaptation, this chapter proposes offline supervised learning through classification for adaptive modulation and coding (AMC) in the MIMO-OFDM system model described in Section 2.3. In offline supervised learning, training sets are generated offline and not during the real-time operation of the system. Although offline supervised learning cannot tune link adaptation to the unique device and environment characteristics of each link, the development of offline algorithms is important to provide a baseline performance evaluation of supervised learning for AMC. Moreover, offline analysis allows for controlled and reproducible simulations that will be necessary to evaluate the efficacy of link quality metrics (feature sets) to be used for training.

Applications of machine learning to AMC in wireless networks have been investigated in the past using supervised learning [61] and stochastic learning models [36, 66] that operate over single antenna channels where the performance is completely parametrized by a single SNR value. While ad-

equate for the systems they model, prior work is difficult to extend since MIMO-OFDM systems experience increased complexity due to coding and interleaving over subcarriers, frequency selective channels, and the need to choose the number of spatial streams along with modulation order and coding rate.

3.1.1 Contributions

In this chapter I complete the system design and analysis of supervised learning for AMC in MIMO-OFDM systems. This includes the creation of a new feature extraction process that produces highly accurate link quality metrics. These new link quality metrics are inspired by a new FER upper bound derived in this chapter. This FER bound shows that the performance of MIMO-OFDM systems is well approximated by the post-processing SNR, without specifying the location of subcarriers. Therefore, the sorted post-processing SNR, over subcarrriers and spatial streams, is used to impose a correlative structure to the post-processing SNR. I exploit this correlation and create a new low-dimension link quality metrics by strategically selecting a subset of the ordered post-processing SNR values. I also propose a new AMC algorithm that uses nearest neighbor classifiers to accomplish link adaptation on a training set of wireless channels with varying frequency selectivity. Finally, I conduct a comprehensive set of simulations in the IEEE 802.11n standard that evaluate the performance of both the proposed supervised learning AMC algorithm and the proposed link quality metrics. These simulations

Table 3.1: Important chapter specific notation.

C	convolutional coding rate
$\mathcal{I}$	transmission parameters sets
W	training set size
i	index for transmission parameter set (MCS)
$\mathbf{z}_w$	feature set $w \in \{0, 1, \ldots, W-1\}$ in training set
$i(w)$	ideal transmission parameter index for ith training set entry
$\mathbf{q}$	feature set query for transmission parameter prediction
p	feature set dimension
P_B	probability of bit error
$\gamma^{(t)}$	the tth smallest ordered post processing SNR
t	ordered subcarrier index, $t \in \{0, 1, \ldots, NN_s - 1\}$
$\mathcal{R}_i$	physical layer data rate for transmission parameter set i
F	frame error rate constraint
$d(\mathbf{z}_w, \mathbf{q})$	distance metric between training data entry and query

show that, not only is AMC through supervised learning capable of highly accurate link adaptation, but also that the proposed ordered post-processing SNR link quality metrics outperform the best known link quality metrics for MIMO-OFDM.

3.1.2 Important Chapter Notation

In addition to the notation presented in the system model of Section 2.3, this chapter uses the notation in Table 3.1.

3.2 Chapter Scope of Machine Learning

Machine learning broadly covers all systems that learn or improve their performance with data observations. In this chapter, machine learning pro-

vides a framework for *classification*. The goal of classification is to place a label, indexed by a non-negative integer or *class*, on an element of interest. To facilitate machine learning and classification, a quantitative measurement, or *feature set* (a vector in real space $\mathbb{R}^p$) for the element is created. A *classifier* maps vectors in $\mathbb{R}^p$ to integers in $\mathbb{N}$. To design a classifier that partitions the feature space in terms of the available classes, empirical inference is accomplished through training data observation. Therefore, a training set that consists of channel realizations associated with the ideal coding rate, modulation order, and number of spatial streams must be available. This training set may be obtained through system measurements or Monte Carlo simulations of the physical layer. To create the training set the FER is simulated/measured for each coding rate, modulation order, and number of spatial streams by repeated transmission of packets. For measurements the resolution of FER in the training set is limited by the coherence time of the channel used to create training sets.

3.3 AMC as Classification

Adaptive modulation and coding in the MIMO-OFDM system model is the process of selecting the QAM modulation order, M, convolutional coding rate, $\mathcal{C}$, and number of spatial streams, N_s, for a given realization of channel state information. In practice, there is only a finite set of allowable M, $\mathcal{C}$, and N_s triplets, where each element of this set is named a *modulation and coding scheme* (MCS). Given that the MCS list is finite, I provide indices to this list

with $i \in \mathcal{I} \subset \mathbb{N}$ where the cardinality of $\mathcal{I}$ is the number of available M, $\mathcal{C}$, and N_s triplets. The index i, hereafter referred to as the *class*, distinctly maps to a realization of M, $\mathcal{C}$, and N_s to define MCS_i. Each class also maps to a single data rate, $\mathcal{R}_i$, provided by the values of M, $\mathcal{C}$, and N_s in MCS_i. Therefore, in terms of classification, AMC is the process of selecting a class i, corresponding to MCS_i, to maximize $\mathcal{R}_i$ under a reliability constraint for different realizations of the channel state. Following convention, due to its importance in higher layer reliability, I consider the FER reliability constraint [12, 90]:

> A class i is only selected if the corresponding FER of MCS_i for the channel $\{\tilde{\mathbf{H}}[n]\}_{n=0}^{N-1}$ with transmit power E_s and receiver noise variance σ^2, defined as $\text{FER}_i(\{\tilde{\mathbf{H}}[n]\}_{n=0}^{N-1}, E_s, \sigma^2)$, is less than or equal to F (i.e. $\text{FER}_i(\cdot) \leq \mathsf{F}$).

Once the set of classes that satisfy the FER constraint are determined, class i is selected to correspond to the MCS_i with highest rate $\mathcal{R}_i$. Therefore, classification selects

$$\arg\max_i \{\mathcal{R}_i : \text{FER}_i(\cdot) \leq \mathsf{F}\} \tag{3.1}$$

to maximize the performance for a given channel realization. If no MCS satisfies (3.1) the most reliable MCS will be selected. If two different classes satisfy the FER constraint and yield the same rate, a rule is needed to determine the class for a given channel. In the next section, Section 3.3.1, which defines the proposed classifier for AMC with supervised learning, the more robust MCS (corresponding to a lower number of spatial streams) is always chosen.

AMC using (3.1) does not necessarily maximize the average data rate over the lifetime of the network. Each realization of channel state maps to the same class throughout the wireless network lifetime (only one class is assumed to be correct). Therefore, AMC selection using (3.1) will always be conservative because every channel realization must meet the FER target. The FER target, however, does not constrain the FER associated with each channel to be less than the target, just the channels in the window of time used to calculate the FER statistic. This conservative MCS selection problem is also found with related work on AMC [12, 90, 94]. The optimal data rate that satisfies the FER constraint is achieved by selecting an entire sequence of MCS values over the window of time used to calculate the FER statistic.

3.3.1 k-Nearest Neighbor AMC

The choice of the classification algorithm depends largely on the structure of classes in the feature space. Unfortunately, for the feature sets chosen in this chapter, namely 1- or 2-dimensional link-quality metrics and ordered post-processing SNR feature sets (to be discussed in the next section), little is known about the functional mapping between feature sets and MCS. As a consequence, non-parametric classification algorithms are preferred [6, 67]. Therefore, I have chosen k-nearest neighbor (k-NN) due to its ability to provide accurate class estimates without knowledge of a functional mapping between the feature sets and the class.

The k-NN system is trained with W distinct realizations of the feature

set and its associated class (i.e. the training set). I define a realization index $w \in \{0, 1, 2, \ldots, W-1\}$ for each feature set in the training set. The feature set corresponding to index w in the training set, $\mathbf{z}_w \in \mathbb{R}^p$, is assigned to a class $i \in \mathcal{I}$ according to (3.1) using the realization of channel state from which $\mathbf{z}_w$ was evaluated. The corresponding MCS for each element of the training set follows the mapping

$$\{\mathbf{z}_w\}_{w=0}^{W-1} \mapsto \{i(w)\}_{w=0}^{W-1} \tag{3.2}$$

where $i(w) \in \mathcal{I}$ is the ideal AMC classification according to (3.1).

Empirical results show that supervised learning does not work well when the set of classes, $\mathcal{I}$, contains MCS with different spatial stream orders. In general, feature sets that are calculated from post-processing SNR measurements (such as all feature sets considered in this chapter) have different meaning depending on the number of spatial streams. Consequently, I apply N_s separate classification procedures over N_s separate training sets. Each training set only considers MCS with the same number of spatial streams. I define the classes corresponding to MCS with a spatial streams as $\mathcal{I}_a$. Likewise I define $\{\mathbf{z}_{w,a}\}_{w=0}^{W_a-1}$ and $\{i(w,a)\}_{w=0}^{W_a-1}$ as the feature set realizations and associated classes for the training set with a spatial streams.

The algorithm for k-NN AMC is as follows and the procedure for supervised learning with N_s spatial streams is illustrated in Fig. 3.1. k-NN AMC predicts the class for a new realization of the channel state information using a query $\mathbf{q} \in \mathbb{R}^p$ representing the feature set for this channel realization. The depth of the search, k, and distance metric $d(\cdot, \cdot)$ remain arbitrary.

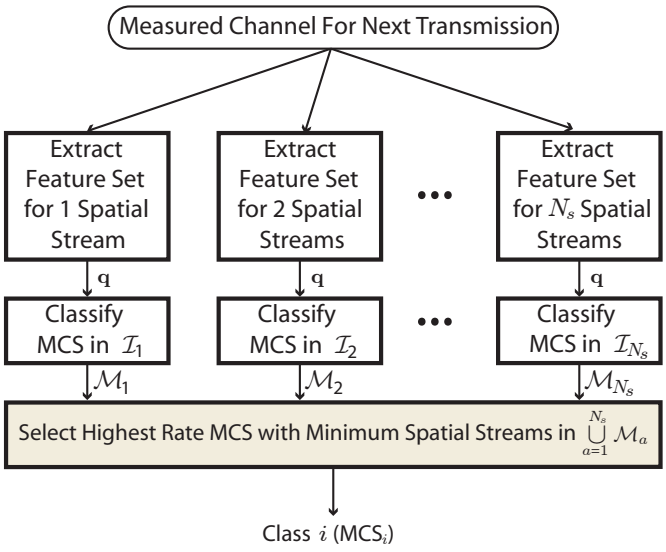


Figure 3.1: Supervised Learning AMC procedure that selects a single MCS across multiple spatial streams

Algorithm *k-NN for AMC*

1. **for** $a \leftarrow 1$ **to** N_s
2. $\quad n_i \leftarrow 0 \; \forall \; i \in \mathfrak{I}_a$
3. $\quad$ **for** $\ell \leftarrow 1$ **to** k
4. $\quad\quad w_\ell \leftarrow \arg\min_w \{d(\mathbf{z}_{w,a}, \mathbf{q}) : w \notin \{w_1, \ldots, w_{\ell-1}\}\}$
5. $\quad\quad n_{i(w_\ell)} \leftarrow n_{i(w_\ell)} + 1$
6. $\quad \mathcal{M}_a = \min\{\arg\max_i \{n_i : i \in \mathfrak{I}_a\}\}$
7. **return** $\min\{\arg\max_i \{\mathcal{R}_i : i \in \cup_{a=1}^{N_s} \mathcal{M}_a\}$

For a feature set query, $\mathbf{q}$, k-NN selects a class i given knowledge of each element in the training set indexed by $w \in \{0, 1, \ldots, W-1\}$ and known to belong to a distinct class i. Implementation of the algorithm uses the distance

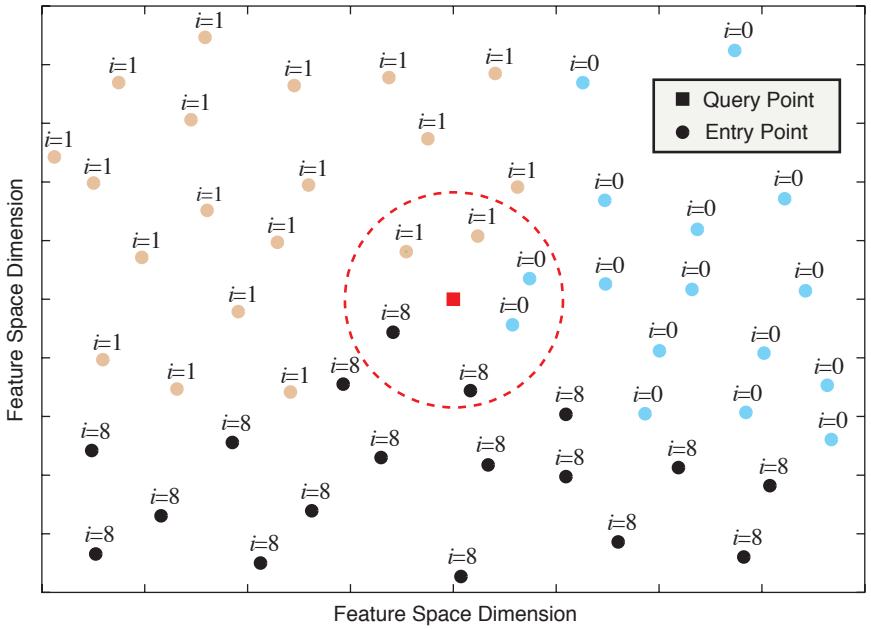


Figure 3.2: 6-NN AMC that assigns MCS_1 since $\mathcal{R}_8 = \mathcal{R}_1 > \mathcal{R}_0$ and $1 < 8$

metric to determine the "neighbors" in the training set. Given a query from a channel observation the classifier searches through the training set to find the k nearest neighbors in terms of the distance metric. The class assigned to the query is the class that occurs most often among the k-nearest neighbors in the training set [24]. The experiments in this chapter use a Euclidean distance metric which implies that all dimensions of the feature set are equally weighted. Other distance metrics may be considered to improve performance, depending on the structure of the classes in the feature space [6].

One unique property of this algorithm is the rule for breaking ties, as illustrated in Figure 3.2. If two distinct classes $i' \neq i''$ both in $\mathcal{I}_a$ return the same number of neighbors to the query for a spatial streams, the tie is broken using the rate. That is, without loss of generality, i' is selected if $\mathcal{R}_{i'} < \mathcal{R}_{i''}$. If i' is selected through classification in $\mathcal{I}_{a'}$ and i'' is selected through classification in $\mathcal{I}_{a''}$ $(a' \neq a'')$ and $\mathcal{R}_{i'} = \mathcal{R}_{i''}$, which often occurs when more than one spatial stream is considered, then the proposed algorithm will always select the lower index, i.e. i' is selected since $a' < a''$. Hence, the proposed algorithm is biased towards lower rate selection and a lower number of spatial streams since empirical studies show this often implies higher communication reliability. A similar selection strategy was chosen in [12].

3.4 Frame Error Rate Analysis

Frame error rate (FER) or the probability that the input binary sequence does not match the estimated binary sequence, is defined

$$\mathrm{FER}(\{\mathbf{H}[n]\}_{n=0}^{N-1}, E_s, \sigma^2) \triangleq \Pr\left[\{b[n_b]\}_{n_b=0}^{N_f-1} \neq \{\hat{b}[n_b]\}_{n_b=0}^{N_f-1} | \{\mathbf{H}[n]\}_{n=0}^{N-1}, E_s, \sigma^2\right] \tag{3.3}$$

for fixed spatial parsing and convolutional coding operations. In practice, assuming the slow fading model, each frame potentially experiences a different channel realization. Hence, the actual frame error rate over the lifetime of the system is

$$\overline{\mathrm{FER}} \triangleq \mathbb{E}_{\{\mathbf{H}[n]\}_{n=0}^{N-1}}\left[\mathrm{FER}(\{\mathbf{H}[n]\}_{n=0}^{N-1}, E_s, \sigma^2)\right] \tag{3.4}$$

where statistical expectation is taken over the distribution of channels. In this chapter, I will refer to FER using (3.3) and for notational simplicity I will abbreviate this with $\mathrm{FER}(\cdot)$.

In this section I investigate FER for MIMO-OFDM systems and arrive at an approximation of FER as a function of post-processing SNR, ordered over subcarriers and spatial streams. This is completed through the analysis of convolutional coding in MIMO-OFDM systems. Because the FER approximation in terms of ordered SNR uses certain assumptions on the QAM constellation, the spatial parsing block size, the constellations on each spatial stream, and the spatial equalization procedure, this section also provides the consequences of relaxing these assumptions.

3.4.1 FER Approximation in MIMO-OFDM with SNR Ordering

For notational convenience let $\mathbf{b}$ be the N_f-dimensional binary vector and $\mathbf{c}$ be the $N_f/\mathcal{C}$-dimensional binary vector formed from the sequences $\{b[n_b]\}_{n_b=0}^{N_f-1}$ and $\{c[n_c]\}_{n_c=0}^{N_f/\mathcal{C}-1}$, respectively. The codeword generator matrix $\mathbf{B}$ is a function of the $q \times r$ generator polynomials characteristic of the convolutional code of rate $\mathcal{C} = q/r$ for coprime $q, r \in \mathbb{N}$ and $r \geq q$. $\mathbf{B}$ relates convolutional encoding with $\mathbf{c} = \mathbf{B}\mathbf{b}$ where all operations (multiplication and addition) are on the binary field $\mathbb{F}_2$. I define an error event when the receiver incorrectly chooses codeword $\hat{\mathbf{c}} = \mathbf{B}\hat{\mathbf{b}}$ and $\mathbf{b} \neq \hat{\mathbf{b}}, \mathbf{c} \neq \hat{\mathbf{c}}$. A well-known performance bound for soft-input Viterbi decoding in terms of P_B, the probability of bit error on decoded bits, states

$$P_B < \frac{1}{q} \sum_{d=d_f}^{\infty} C_d P_d. \tag{3.5}$$

given d_f as the minimum distance error event, C_d as the number of source bit errors due to d-distance error events, and P_d as the probability of d-distance error events [96]. C_d is a function of $\mathbf{B}$ and P_d is additionally a function of $\{\mathbf{H}[n]\}_{n=0}^{N-1}$, E_s, and σ^2.

Consider $\gamma^{(t)} \in \{\gamma[1,0], \ldots, \gamma[N_s, N-1]\}$ as the $(t+1)$th smallest post-processing SNR from (2.22) for all subcarriers and spatial streams such that

$$\gamma^{(0)} \leq \gamma^{(1)} \leq \cdots \leq \gamma^{(N_s N-1)} \tag{3.6}$$

where $\gamma^{(0)} = \min_{a,n} \gamma[a,n]$ and $\gamma^{(N_s N-1)} = \max_{a,n} \gamma[a,n]$ for $a \in \{1, 2, \ldots, N_s\}$, $n \in \{0, 1, \ldots, N-1\}$. Next, I derive the FER bound in the MIMO-OFDM

system model

$$\mathrm{FER}(\cdot) < 1 - \left(1 - \frac{1}{q}\sum_{d=d_f}^{d_L} C_{d,L} \prod_{l=1}^{d} Z\left(\gamma^{(l)}\right)\right)^{N_f} \tag{3.7}$$

with $C_{d,L}$ as the number of source bit errors due to d-distance error events of length at most L, d_L as the maximum distance error event with length $\leq L$, and $Z(\gamma) = \exp(-\gamma)$. This FER approximation enables performance characterization without explicitly defining the spatial parsing process.

3.4.2 Derivation

For length N_f frames (containing N_f bits), FER $\triangleq 1-(1-P_B)^{N_f}$ where P_B is the probability of bit error in the decoded bits. Below I examine bounds on P_B that eventually lead to an approximation of P_B for the MIMO-OFDM system model.

P_B **for SISO-OFDM**: Consider single-input-single-output OFDM (i.e., $N_s = 1$), or SISO-OFDM, in block fading with binary phase-shift keying (BPSK). I characterize interleaving with the permutation $\pi : \{0, 1, \ldots, N_f/\mathcal{C}-1\} \rightarrow \{0, 1, \ldots, N_f/\mathcal{C}-1\}$ that acts on the indices of the binary sequence. Let the notation $\pi(n)$ denote the index of the interleaved sequence corresponding to index n of the sequence before interleaving. For an AWGN channel with BPSK constellations and SNR γ, the Viterbi bound states that $P_B < 1/q \sum_{d=d_f} C_d \exp(-d\gamma) = 1/q \sum_{d=d_f} C_d Z(\gamma)^d$ where $Z(\gamma) \triangleq \exp(-\gamma)$ [96]. This bound can also be directly translated to flat fading by incorporating the channel gain into the noise variance. SISO-OFDM observes N

fading channels corresponding to each subcarrier n, each parameterized by $\gamma[n] \triangleq E_s|\mathbf{H}[n]|^2/\sigma^2$, the SISO SNR. Therefore for SISO-OFDM with S_d as the set of all possible codewords generated by $\mathbf{B}$ with weight d I have

$$P_B < \frac{1}{q} \sum_{d=d_f}^{\infty} \sum_{\mathbf{c} \in \mathsf{S}_d} \prod_{l=1}^{d} Z\left(\gamma\left[\mathrm{mod}\left(\pi(\ell_l^{(\mathbf{c})}), N\right)\right]\right) \tag{3.8}$$

where $\{\ell_1^{(\mathbf{c})}, \ell_2^{(\mathbf{c})}, \ldots, \ell_d^{(\mathbf{c})}\}$ is the set of indices for codeword $\mathbf{c}$ that reference non-zero bits and $\mathrm{mod}(n, N)$ returns the remainder of $\frac{n}{N}$.

P_B for MIMO-OFDM with Zero-Forcing Receivers: For MIMO-OFDM with linear receivers, it is straightforward to calculate per-stream SNR values for each subcarrier as seen in equation (2.22). By construction, ZF equalization completely removes inter-stream interference and Gaussianity is preserved in the effective noise vector. For MIMO-OFDM the interleaving function is replaced by a spatial parsing function s. Assuming a general spatial parsing function with $L \geq N_f/\mathcal{C}$, let $s : \{0, 1, \ldots, N_f/\mathcal{C}-1\} \rightarrow \{1, 2, \ldots, N_s\} \times \{0, 1, \ldots, N_f/(N_s\mathcal{C}) - 1\}$ represent the spatial parser in Fig. 2.9. Furthermore $s_1(n) \in \{0, 1, \ldots, N_s\}$ is equal to the spatial stream that the coded bit indexed by n is mapped to. Likewise $s_2(n) \in \{1, 2, \ldots, N_f/(N_s\mathcal{C})\}$ is equal to the parsed bit index in spatial stream $s_1(n)$. Fig. 3.3 demonstrates the operation of these functions. Using similar arguments as before,

$$P_B < \frac{1}{q} \sum_{d=d_f}^{\infty} \sum_{\mathbf{c} \in \mathsf{S}_d} \prod_{l=1}^{d} Z\left(\gamma\left[s_1(\ell_l^{(\mathbf{c})}), \mathrm{mod}\left(s_2(\ell_l^{(\mathbf{c})}), N\right)\right]\right). \tag{3.9}$$

P_B Bound with Ordered SNR: Practical interleavers process L bits consecutively at a time to limit the complexity and processing delay for the

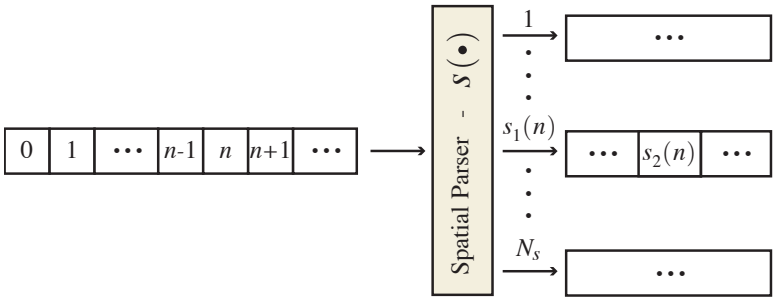


Figure 3.3: Spatial parsing function notation

interleaving process. Past research on the truncation length with Viterbi decoding has shown only T_b consecutive bits are necessary to capture virtually all error events that occur in practice. If $L \leq N_s N$ and $T_b \leq L$, which is sufficient for the convolutional codes in IEEE 802.11n and IEEE 802.16e with generators $(171, 133)_8$ [72],

$$\max_{\mathbf{c}\in\mathsf{S}_d} \left\{ \prod_{l=1}^{d} Z\left(\gamma\left[s_1(\ell_l^{(\mathbf{c})}), \bmod\left(s_2(\ell_l^{(\mathbf{c})}), N\right)\right]\right) \right\} \leq \max_{\substack{a_1,a_2,\ldots,a_d \\ n_1,n_2,\ldots,n_d}} \left\{ \prod_{l=1}^{d} Z\left(\gamma[a_l, n_l]\right) \right\} \tag{3.10}$$

where $a_l \in \{1, 2, \ldots, N_s\}$, $n_l \in \{0, 1, \ldots, N-1\}$ and $(a_l, n_l) \neq (a_\alpha, n_\alpha)$ for $\alpha \neq l$. The bound allows $d \leq d_L \leq L$ where d_L is the maximum codeword weight in L-length error events. Assuming no error events of length $> L$ or codeword errors of weight $> d_L$ I conclude

$$P_B < \frac{1}{q} \sum_{d=d_f}^{d_L} C_{d,L} \prod_{l=1}^{d} Z\left(\gamma^{(l)}\right) \tag{3.11}$$

given $C_{d,L}$ equals the number of information bit errors due to weight d codeword errors with error event lengths $< L$. Simulations have shown that this bound is tight (average SNR gap ≤ 0.5 dB) for $P_B \leq 10^{-4}$ in most frequency

selective channels which implies that FER bounds are tight for reasonable frame lengths ($N_f \geq 1000$) and FER constraints ($\mathsf{F} \leq 0.1$).

3.4.3 Exceptions

Spatial Parsing (Interleaving) Block Length: For (3.7) I assume a spatial parsing block length $L \leq N_s N$ as in IEEE 802.11n. Other MIMO-OFDM standards, such as 3GPP-LTE and WiMax, allow for the interleaving block length to be larger than $N_s N$. When $L > N_s N$ the block size for the spatial parsing function exceeds the total number of subcarriers for all spatial streams, hence the worst case post-processing SNR approximation used to derive (3.7) in Appendix 3.4.2 includes redundant ordered post-processing SNR terms. In either case, as long as L exceeds the truncation length of the convolutional code, the FER approximation is still parametrized by the ordered post-processing SNR over all subcarriers and spatial streams.

***M*-QAM Constellations**: The FER bound in (3.7) assumes BPSK constellations. The bounds in (3.8) and (3.9) that lead to this approximation can be modified to include QAM constellations of general order, assuming Gray coding and equal probability of bit errors if an incorrect constellation point is selected. The probability of bit-error for these general constellations do not lead to simple definitions of P_d and convenient expression in terms of $Z(\cdot)$. Nevertheless, the bounds for M-QAM constellations are similarly a function of the SNR per subcarrier. Hence, different constellations do not change the parameters needed to determine FER (for an example see [63, 75]).

Non-uniform Energy and Constellations: Non-uniform energy allocation over subcarriers and spatial streams is allowed in the system model, although not explicitly provided. Different power levels over subcarriers and spatial streams may be embedded in the precoding matrices $\mathbf{F}[n]$ for each subcarrier. Non-uniform constellations, however, are not covered under the current system model. Some systems, such as the optional MCS modes of IEEE 802.11n, allow for mixed QAM dimensions over different spatial streams [48]. Non-uniform constellations will further increase the complexity of FER bounds since P_d is determined by the probability of bit error for different sized constellations.

Alternative Equalization: The FER approximation in (3.7) assumes ZF equalization, although other spatial equalization methods may be implemented in practice such as minimum mean-square-errror (MMSE) linear receivers and ML detection over spatial streams (and near-ML approximation). MMSE equalizers do not produce Gaussian inter-stream interference plus noise terms, however, Gaussian approximations are often used [82]. Consequently, I expect the FER approximation in (3.7) to apply directly to MMSE receivers with post-processing SNR calculations from (2.22). Post-processing SNR, however, is not easily defined for full ML spatial equalization. One solution is to use the singular values of the channel matrix in the FER bounds since they characterize the post-processing SNR of ML spatial equalization when ideal SVD precoding is used at the transmitter [74].

3.5 Feature Set Selection

The choice of an appropriate feature space is critical due to the phenomenon in supervised learning known as "the curse of dimensionality" [5]. That is, each dimension that is added to the feature space requires increasingly more elements in the training set to uniformly sample the feature space. Therefore in IEEE 802.11n systems with 2 transmit and 2 receive antennas, using the full channel in the feature space would require 416 dimensions (52 data subcarriers $\times$ 2 transmit RF chains $\times$ 2 receive RF chains $\times$ 2 dimensions of complex channel), resulting in complete failure of classification with training sets of reasonable size. It is necessary to find a low-dimension feature set that captures the essence of (3.1) in the these 416 dimensions.

Prior work on AMC for MIMO-OFDM has resulted in several one- or two-dimensional link-quality metrics extracted from channel state that attempt to parametrize FER for look-up-table based AMC. Empirical results have shown, however, that these link quality metrics do not contain enough information about the quality of the channel per-subcarrier and spatial stream in frequency and spatial selective channels to perform MIMO-OFDM AMC. This is confirmed through the simulations of Section 3.6 in this chapter. I proceed with the creation of a feature set for supervised learning in MIMO-OFDM based on post-processing SNR, ordered over subcarriers and spatial streams.

Given (3.7), I know the set of ordered SNR, $\left\{\gamma^{(0)}, \gamma^{(1)}, \ldots, \gamma^{(N_s N-1)}\right\}$, uniquely parametrizes the approximate FER in MIMO-OFDM systems. Un-

fortunately, the dimension of the set of ordered SNR, $N_s N$, is still too large to represent the feature set for supervised learning. One key observation, however, is the significant correlation between the ordered SNR values due to the sorting process and the bounds on achievable SNR. I exploit this correlation through a simple low-dimensional mapping, $\mathcal{P}$, from the $N_s N$-dimensional ordered post-processing SNR set to the p-dimensional ordered post-processing SNR feature set, defined

$$\mathcal{P}\left(\left\{\gamma^{(0)}, \gamma^{(1)}, \ldots, \gamma^{(N_s N-1)}\right\}\right) \triangleq \left\{\gamma^{(n_1)}, \gamma^{(n_2)}, \ldots, \gamma^{(n_p)}\right\} \tag{3.12}$$

with indices $n_1, n_2, \ldots, n_p \in \{0, 1, \ldots, N_s N - 1\}$. Essentially, to calculate the feature space I first sort the post-processing SNR over all subcarriers and spatial streams and then select p of these sorted SNR values. In Section 3.6 I show a procedure for discovering the best indices, $\{n_1, n_2, \ldots, n_p\}$, for a given training set. If the correct subcarrier indices are chosen, an accurate approximation of the set $\left\{\gamma^{(0)}, \gamma^{(1)}, \ldots, \gamma^{(N_s N-1)}\right\}$ is obtained through only p dimensions.

3.6 k-NN AMC for IEEE 802.11n

To illustrate the performance of the AMC algorithm I follow the MCS specifications from IEEE 802.11n [43] with 2 receive antennas and 2 transmit antennas. Consider the first 16 MCS for the IEEE 802.11n physical layer highlighted in Table 3.2. Each class i for MCS_i defines a convolutional coding rate, QAM modulation order, M, and number of spatial streams, N_s, to determine the physical layer data rate $\mathcal{R}_i$. First, I determine a suitable feature set

Table 3.2: IEEE 802.11n MCS List

MCS_i	N_s	M	Code Rate	$\mathcal{R}_i$
$i = 0$	1	2	1/2	6.5 Mbps
$i = 1$	1	4	1/2	13.0 Mbps
$i = 2$	1	4	3/4	19.5 Mbps
$i = 3$	1	16	1/2	26.0 Mbps
$i = 4$	1	16	3/4	39.0 Mbps
$i = 5$	1	64	2/3	52.0 Mbps
$i = 6$	1	64	3/4	58.5 Mbps
$i = 7$	1	64	5/6	65.0 Mbps
$i = 8$	2	2	1/2	13.0 Mbps
$i = 9$	2	4	1/2	26.0 Mbps
$i = 10$	2	4	3/4	39.0 Mbps
$i = 11$	2	16	1/2	52.0 Mbps
$i = 12$	2	16	3/4	78.0 Mbps
$i = 13$	2	64	2/3	104.0 Mbps
$i = 14$	2	64	3/4	117.0 Mbps
$i = 15$	2	64	5/6	130.0 Mbps

dimension for ordered SNR feature sets, the best subcarrier indices for that dimension, and the best k to select for channels of varying selectivity. Second, I analyze the system performance (FER and spectral efficiency) of k-NN AMC using the value of p, the indices, and the value of k determined from the first part of the simulation.

3.6.1 Simulation Assumptions

The system used for simulation studies in this section assumes the following.

- *Fixed Frame Length*: All frames are 128 bytes in length. FER is a

function of the number of bits per frame. In practice many different frame lengths are used depending on the traffic source. For simplicity in simulations this chapter does not consider different frame lengths.

- *Perfect Synchronization*: I neglect the pilot tones and training sequence. Errors in synchronization can lead to frame errors that are not associated with improper selection of the MCS. Therefore, channel frequency offset and symbol timing are not considered.

- *Perfect Channel Knowledge*: I assume that k-NN AMC knows the effective channel at the receiver perfectly. Channel estimation error results in two forms of performance degradation: symbol equalization error and feature set extraction error. Fortunately, classification error due to equalization performance degradation is minimal since the training set is able to capture this effect. Feature set extraction error, however, will result in misclassified features both in the training set and during the k-NN AMC process. Simulations and measurements have shown that in an interference-free environment feature set extraction error (primarily due to thermal noise) is mollified by the SNR ordering operation as long as the extreme values of SNR (largest or smallest) are not used to characterize the SNR profile.

- *Zero-Forcing Linear Equalization*: Equalization uses a soft-output ZF structure in the frequency domain to compute log likelihood ratios for a Viterbi detector. Other equalization procedures are compatible with

subcarrier ordering feature sets by redefining the post-processing SNR ratio.

- *No Space-Time Processing*: No space-time processing techniques such as space-time block coding, beamforming, or precoding occur during simulation. With space-time processing techniques, not only does the MCS have to be determined, but additionally a decision is made with regard to which precoding matrix or diversity technique to use.

The basic parameters for this IEEE 802.11n-based study of k-NN AMC are summarized in Table 3.3. In IEEE 802.11n the number of data subcarriers (discrete Fourier transform (DFT) size minus pilot subcarriers and guard tones) $N_{ds} = 52$ for 20 MHz channels. Therefore, the complete ordered post-processing SNR set is $\left\{\gamma^{(0)}, \gamma^{(1)}, \ldots, \gamma^{(N_s \times 52-1)}\right\}$.

In Section 3.6.2 I determine the dimension of the ordered SNR feature set, the indices in the ordered post processing SNR profile to represent the ordered SNR feature set, as well as the best value of k in k-NN. In these simulations I use three channel distributions, all generated from uniform power delay profiles: (1) frequency-flat channels (1-tap) (2) moderately frequency selective channels (3-tap) (3) very frequency selective channels (8-tap). Eight taps represent the maximum frequency selectivity since the IEEE 802.11n cyclic prefix length is either 8 or 16 time-domain symbols in length for 20 MHz channels. For each of these 3 channel distributions, I create a training set and test set each with 1000 realizations of the $N_r \times N_t$ channel matrix for each tap.

Table 3.3: Simulation Parameters for k-NN AMC in IEEE 802.11n

Property	Value
Frequency Flat Channel	1 tap $\in \mathbb{C}^{N_r \times N_t}$ each single-tap matrix element $\sim \mathcal{CN}(0,1)$
Moderately Selective Channel	3 taps, each tap $\in \mathbb{C}^{N_r \times N_t}$ each tap matrix element $\sim \mathcal{CN}(0,1)$
Very Selective Channel	8 taps, each tap $\in \mathbb{C}^{N_r \times N_t}$ each tap matrix element $\sim \mathcal{CN}(0,1)$
Transmit Energy (above sets only)	$10\log_{10}(E_s/\sigma^2) \in \{1, 2, \ldots, 28\}$
Realizations (W) (above sets only)	$1000 \times 28 = 2.8 \times 10^4$
Performance Analysis Channel	IEEE 802.11n Channel Model B 5000 Realizations total - 5 to 50 m
Classes (i)	$i \in \{0, 1, \ldots, 15\}$ - MCS in IEEE 802.11n
FER Target (F)	0.1 (10% FER)
$N_r \times N_t$	$2 \times 2 \Rightarrow N_s \in \{1, 2\}$
k	25
$d(\cdot)$	Euclidean distance

Every matrix element for every tap is complex-Gaussian distributed with zero mean and unit variance. Furthermore, each of the 1000 channel realizations in the training and test sets are combined with 28 values of E_s/σ^2, resulting in 2.8×10^4 feature realizations in each training and test set. The simulations are completed by first training the k-NN classifier on the training set for each of the 3 channels. The classifiers are then verified on a test set for each of the 3 channels. The test and training sets observe the same statistical distribution although they each contain distinct realizations of the channel.

3.6.2 Selecting the Subcarrier Ordering Indices and k

To determine the best indices for ordered post-processing SNR feature sets I perform a brute force search over all possible ordered subcarrier feature sets up to four dimensions. This is completed as follows for a fixed feature space dimension, p, fixed number of spatial streams, and fixed neighbor size, k.

1. Given k and p select a set of ordered subcarrier indices not yet simulated.

2. For each channel and E_s/σ^2 realization in the training set compute the post-processing SNR in (2.22), sort according to (3.6), and select the ordered post-processing SNR values corresponding to the indices in 1. to construct the feature set, $\mathbf{z}$.

3. Using (3.1) associate $\mathbf{z}$ with an MCS label after calculating the FER for each MCS through Monte Carlo simulation with the channel realization

associated with $\mathbf{z}$.

4. Repeat 2. through 3. until all channels in the training set are exhausted.

5. For each channel and E_s/σ^2 realization in the test set compute the post-processing SNR in (2.22), sort according to (3.6), and select the ordered post-processing SNR values corresponding to the indices in 1. to construct a query, $\mathbf{q}$.

6. Find the k-nearest neighbors to the query in the training set. Apply the MCS label to the query according to the k-NN AMC algorithm in Section 3.3.1.

7. Repeat 5. through 6. until all channels in the test set are exhausted.

8. Count all errors k-NN AMC made in 5. and 6. for predicting the MCS of queries in the test set.

9. Repeat 1. through 8. until all possible ordered subcarrier indices are simulated. Choose the subcarrier indices with the lowest number of errors for each fixed dimension, p.

Table 3.4 shows the results to this search for the 3 and 8 tap channel distributions with $k = 25$. The 1-tap channel was not included; any subcarrier feature set that includes post-processing SNR from each spatial stream will perform the same. Only four 4 dimensions are shown in Table 3.4 since more than 4 dimensions did not improve performance substantially.

Table 3.4: Best Subcarrier Ordering Sets for k-NN AMC in IEEE 802.11n (Brute Force Search)

N_s	p	3 Tap $(n_1, n_2, \ldots, n_p)$	8 Tap $(n_1, n_2, \ldots, n_p)$
1	1	(13)	(17)
1	2	(8,31)	(10,31)
1	3	(6,16,35)	(6,17,39)
1	4	(5,10,23,40)	(5,12,24,38)
2	1	(25)	(27)
2	2	(11,52)	(14,46)
2	3	(7,24,61)	(7,23,56)
2	4	(6,13,24,56)	(6,8,27,50)

Using $(n_1, n_2, \ldots, n_p)$ found in Table 3.4 with $p = 4$ to create ordered SNR feature sets, I also investigate the performance of k-NN AMC for different values of k. Fig. 3.4 shows the performance through the percentage of correct MCS labels assigned in the test set with the 1, 3, and 8 tap channels when k is varied from 5 to 45. Initially, there is a significant performance improvement for increasing k, but around $k = 25$, the performance levels out for increasing values of k in any channel set. Generally, increasing the size of k reduces errors by limiting the effect of mislabeling that occurs during training set construction. Additionally, classification errors occur when queries from the test set are not close in the distance metric to any feature sets corresponding to channel realizations in the training set. Large k values can help to reduce the effect of such occurrences. Due to the limited size of training sets, however, increasing k past a certain point will start to produce errors.

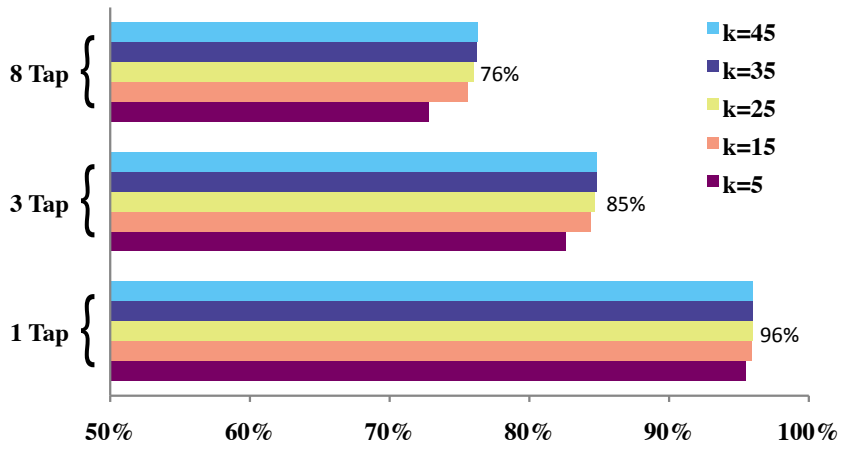


Figure 3.4: Correct classification percentages in the test set of 1, 3, and 8 tap channels using the best 4-dimension feature sets from Table 3.4. The different values of k are displayed on the clustered bar graph. The order of the color code in each bar graph cluster matches the order in the legend.

3.6.3 Comparison to Alternative Feature Spaces

For completeness and validation of the subcarrier ordering feature set, I compare k-NN AMC with different feature sets. I use the indices from Table 3.4 to construct 4-dimensional ordered post-processing SNR feature sets in the 1, 3, and 8 tap channels. Fig. 3.5 compares the performance (through the percentage of MCS values correctly selected) of ordered post-processing SNR feature sets to single-dimensional link quality metrics. Each of these features is used in prior work as link-quality metrics to map from MCS to FER. Absent from this list are mutual information derived metrics. Mutual information metrics depend explicitly on M, the QAM modulation order. Hence, this link-quality metric is not applicable to the k-NN AMC procedure that uses a

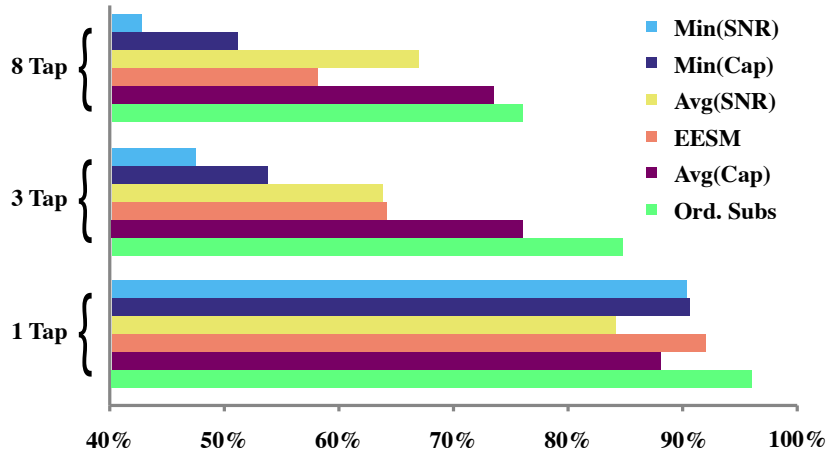


Figure 3.5: Correct classification percentages in the test set with $1, 3$, and 8 tap channels and $k = 25$ for several feature sets. The ordered post-processing SNR feature sets use the best indices from Table 3.4 with $d = 4$. The order of the color code in each bar graph cluster matches the order in the legend.

single feature realization for all MCS within a single spatial stream. EESM features, when used in look-up-tables, are tuned for each MCS by adjusting the β parameter between $\beta = 1$ and $\beta = 25$ [59]. In this simulation, however, β tuning is unnecessary since I do not need the FER associated with each EESM metric to be aligned. During testing, tuning β between 1 and 25 did not change the number of classification errors.

The performance of ordered post-processing SNR feature sets in k-NN AMC outperforms all other potential feature sets for all three channel distributions. This is somewhat surprising because all of the single-dimensional link-quality metrics should uniquely parametrize frequency-flat channels. However, in the MIMO-OFDM system with $N_s > 1$ spatial streams, there are multiple metrics to consider, namely the post-processing SNR for each of N_s streams in

the frequency flat channel. Therefore, single-dimensional link-quality metrics do not uniquely parametrize FER and suffer from classification errors.

Also notable is the performance of the different feature sets in 8-tap channels. The gap between the performance of subcarrier ordering and the alternative feature sets has narrowed. The performance of average capacity feature sets approaches ordered post-processing SNR feature sets. This is due to the curse of dimensionality. If more taps are observed in the channel (with a uniform power delay profile), more frequency selectivity is observed. Hence, more taps yield more diversity in the ordered post-processing SNR profile. This is confirmed by observing the difference in performance between 3 and 8 tap channels for the feature sets based on 1-dimensional link quality metrics.

3.6.4 Frame Error Rate and Spectral Efficiency

Next, I measure the performance of k-NN AMC as a function of feature set. Again, distinct training and test sets are constructed, this time with 5000 channel realizations in IEEE 802.11n channel model B over a transmission distance of $5-50$ meters [93]. k-NN is used with $k = 25$ to meet a target FER of 10% ($\mathsf{F} = 0.1$). Subcarrier ordering feature sets use four dimensions with indices chosen from the 8 tap channel scenario in Table 3.4. Fig. 3.6 shows the FER as a function of SNR averaged over subcarriers and spatial streams when k-NN AMC is implemented with several feature sets. For clarity in the plots, minimum SNR and average SNR feature sets were not included because their performance was inferior to the remaining feature sets. The ideal k-NN AMC

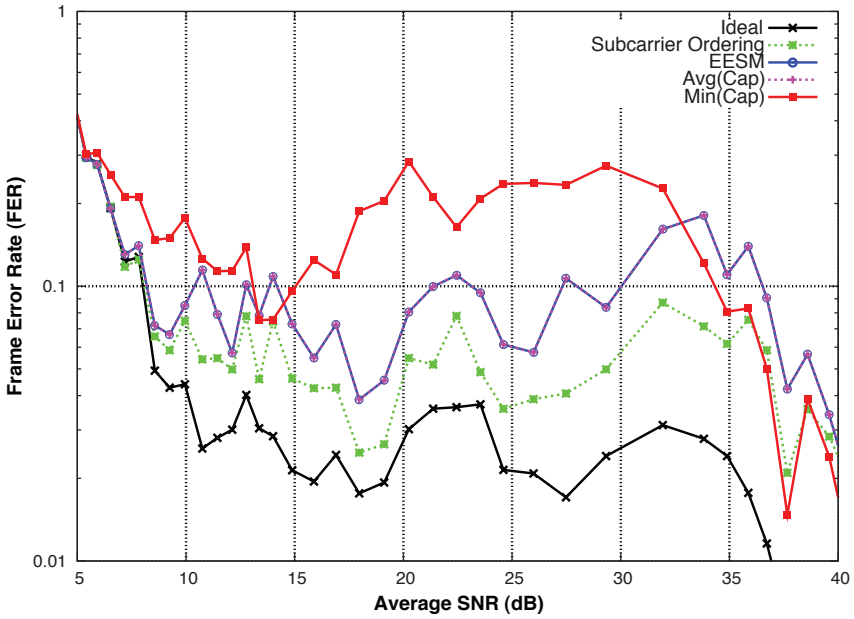


Figure 3.6: Frame error rate as a function of average SNR (over subcarriers and spatial streams from equation (2.22)) when k-NN AMC is applied with different feature sets in IEEE 802.11n channel model B. FER target is 10%. Average capacity and EESM feature sets display the same frame error rate. Since average SNR is not controlled in the simulation, sample points on the x-axis are the result of binning and averaging FER over all samples in the bin.

procedure follows the adaptation criterion in (3.1). The ragged nature of the FER curves is an artifact of AMC with an FER constraint. To meet the FER constraint and optimize data rate, each average SNR is a mixture of different MCS in different proportions. I expect this ragged behavior regardless of the number of iterations/samples in the simulation.

The FER constraint cannot be met when average SNR is less than ≈ 7 dB. This occurs when MCS_0 (the most robust MCS) does not observe the 10% FER constraint. Above 7 dB the classifier determines which feature sets allow k-NN AMC to meet the FER constraint. Although subcarrier ordering does not exactly match the FER performance of ideal MCS selection in (3.1), it is the only feature that does not exceed the 10% FER threshold. The next best feature sets were EESM and average capacity, which both showed identical performance for k-NN AMC in IEEE 802.11n channel model B.

The corresponding spectral efficiency of k-NN AMC in IEEE 802.11n channel model B is shown in Fig. 3.7. The spectral efficiency for each channel realization is defined as the data rate of the MCS selected divided by the bandwidth of the transmission. Therefore, the maximum spectral efficiency corresponds to MCS_{15} with 130 Mbps/20 MHz = 6.5 bps/Hz. Similarly the minimum spectral efficiency is .375 bps/Hz for MCS_0. In Fig. 3.7 spectral efficiency is averaged over the all channels in the test set corresponding at each average SNR. Following the definition in [12], if the FER constraint is violated, the spectral efficiency is 0 bps/Hz. The spectral efficiency of k-NN AMC with ordered post-processing SNR feature sets distinctly outperforms

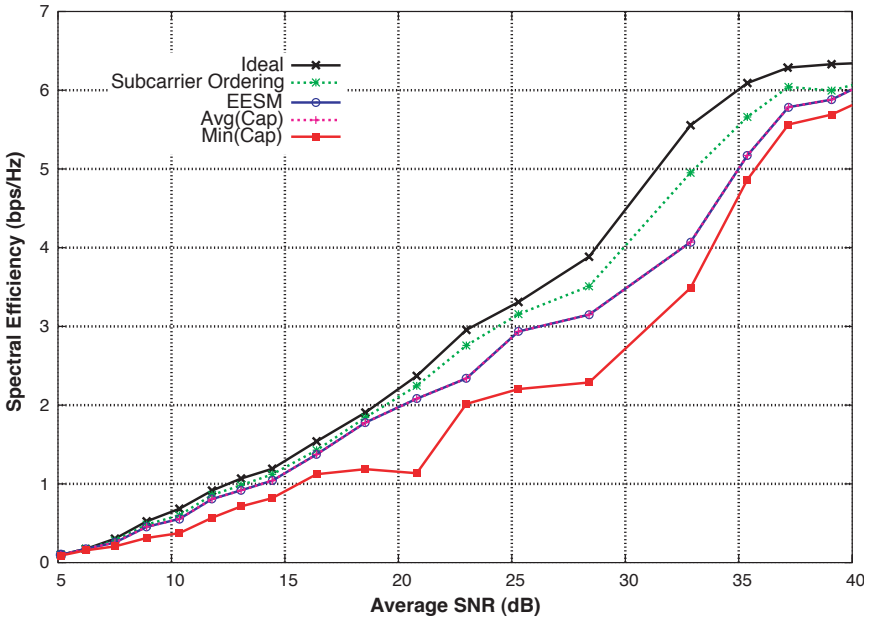


Figure 3.7: Spectral efficiency as a function of average SNR (over subcarriers and spatial streams using equation (2.22)) when k-NN AMC is applied with different feature sets in IEEE 802.11n channel model B. FER target is 10%. Average capacity and EESM feature sets display the same spectral efficiency. Since average SNR is not controlled in the simulation, sample points on the x-axis are the result of binning and averaging FER over all samples in the bin.

the best performing feature sets based on link-quality metrics by as much as 1 bps/Hz.

3.6.5 k-NN AMC Complexity

This chapter is not primarily concerned with complexity issues since I am attempting to provide a baseline for AMC with machine learning by validating the efficacy of k-NN with feature sets extracted from ordered subcarrier SNR. Nevertheless, a complexity discussion is relevant to demonstrate implementation challenges. There are two significant components of k-NN AMC in terms of complexity: the nearest neighbor search and the memory space of training data. From a wireless device design perspective, the memory dedicated to k-NN is the major disadvantage since a large quantity of reserved memory corresponds to valuable silicon space on single chip solutions. For example, the training set of k-NN AMC for the experiments of this chapter require ≈ 1 MB for floating point features. Fortunately, memory space can be significantly reduced. For example, with the ordered SNR feature sets considered in this chapter, there is only a finite range of potential values in each dimension due to constraints on feasible transmitted power and receiver sensitivity. Moreover, high numerical precision (floating point) is not needed for each dimension.

A brute force k-NN search displays complexity $O(pW)$. Brute force searching, however, does not exploit any structure in the information databases. A number of k-NN algorithms reduce the complexity of searches by restruc-

turing databases into a grid. It is well known that grid structuring algorithms with $O(pW \log W)$ complexity to reassign the database memory can reduce the *expected* search complexity to $O(\log W)$ [29]. For AMC, database grid structuring is not latency critical since it can be performed offline. Although the expected complexity is favorable for k-NN with structured databases, the worst case search complexity does not follow $O(\log W)$ [3]. Another means of reducing complexity in the k-NN search is to accept an error tolerance ϵ in the nearest neighbor. Using this principle I can guarantee search order complexity of $O((\alpha + kp) \log W)$ for some $\alpha \leq p\lceil 1 + 6p/\epsilon \rceil^p$ [3].

3.7 Summary

Convolutionally-coded, bit-interleaved MIMO-OFDM systems provide challenging AMC in spatial- and frequency-selective channels. In this chapter I design supervised learning classifiers through nearest neighbor algorithms to provide a framework for accurate AMC in MIMO-OFDM. First, I derive a FER bound for convolutionally-coded, bit-interleaved MIMO-OFDM systems. This bound shows that MIMO-OFDM performance is insensitive to the SNR location, i.e., a sorting operation across subcarriers and spatial streams does not significantly impact the post-processing SNR as a link quality metric. This creates substantial correlation that may be exploited to provide low-dimension feature sets amenable with classification for AMC. These feature sets were shown to outperform the best known link quality metrics for AMC in IEEE 802.11n through a comprehensive simulation analysis.

Chapter 4

Online Supervised Learning for Link Adaptation

4.1 Introduction

In Chapter 3 I showed that offline supervised learning applied to link adaptation through classification can take advantage of new multi-dimensional link quality metrics to improve adaptation accuracy. A major challenge with practical link adaptation still persists. Both look-up-tables and offline supervised learning are not updated in real-time and not tuned to the unique characteristics of each transmit/receive device pair. Hence, offline link adaptation cannot capture all factors that determine error rate and will inevitably adapt incorrectly in practical deployments that experience nonlinearities and non-Gaussian additive noise [22, 37]. The observation of evolutionary impairment behavior, for example evolving nonlinearities with temperature changes in devices [100] or evolving co-channel interference statistics with mobility [80], further exacerbates this problem. Even when it is possible to tune offline link adaptation to each transmit/receive device pair through a startup calibration procedure, the behavior of link adaptation must react to its environment. Fortunately, supervised learning provides a flexible algorithm structure that is amenable to online operation. If training data is captured in real-time, system

model evolution can also be captured. With online learning, link adaptation is optimized to the unique and potentially dynamic characteristics of each wireless device.

The other alternative solution for optimizing link adaptation in practical systems is a more comprehensive system model that captures the effects of nonlinearities and non-Gaussian noise effects for offline analysis. I have not chosen this approach for two primary reasons. First, the complexity of the LTI-AWGN model for offline analysis has already resulted in significant tractability issues in modern wireless networks with MIMO-OFDM links. Deriving expressions of performance with more complex link models appears to be exceedingly difficult. Second, adding these factors to the system model also increases the complexity of a generative model for Monte Carlo simulations. Models are necessary to capture the effects of power amplifier nonlinearity, mixer nonlinearity, co-channel interference, fixed-point arithmetic, I/Q imbalance, phase noise, non-ideal filters, and more. Finding the appropriate link adaptation response in terms of all these effects leads to large engineering design time. Such effort is likely not justified for link adaptation which, although important, is only one component of digital wireless communication. As a consequence, I have chosen online learning as the preferred method of link adaptation in practice.

4.1.1 Contributions

In this chapter I propose a comprehensive set of algorithms as well as the system architecture to enable online AMC through supervised learning. First, I propose online AMC with database learning through nearest neighbor classifiers. To capture information about the frame error rate constraint from received frames transmitted with a single MCS, I take advantage of the maximum likelihood estimate of the frame error class probability for each MCS in the neighbor discovery process. Because training data is collected in real-time and because databases are finite-sized, feature diversity preservation techniques are necessary to prevent conservative AMC. I propose the use of density bounding algorithms when databases are updated to maintain feature set richness and maintain a sufficient and relevant history of past channel observations. Moreover, algorithms for exploration are provided as a means to obtain diverse information about the wireless channel when it is not already known. Protocol guidelines for the operation of exploration are provided for scenarios of low network utilization and when conservative link adaptation is observed. Experimental measurements of database learning with the Hydra IEEE 802.11n physical layer show that database learning outperforms offline link adaptation algorithms as well as competing online AMC algorithms, namely auto-rate fallback (ARF). Further, simulations show that feature set diversity preservation and exploration are vital to the operation of online AMC with supervised learning by preventing conservative AMC performance. I also provide a comprehensive complexity discussion of database

learning, which shows that online AMC database learning may not be feasible for highly-integrated single chip solutions.

To reduce the complexity of online AMC through supervised learning I propose the use of support vector machines (SVM) to enable machine learning classification for fast AMC. Although, SVMs were applied to fast AMC successfully in the past [101, 105], the data set used to train these SVMs was constructed offline using frame error rate estimates of fixed channel realizations. Online AMC through SVMs, however, is not straightforward since the posterior probability of classes is not immediately available from the output of SVM inference functions, leading to difficulty when addressing frame error rate constraints. Consequently, I propose regression of the inference function outputs on the Platt sigmoid to extract posterior probability for the frame error class of each MCS. Through simulation, by demonstrating a small set of support vectors, I show that the online SVM fast AMC algorithm results in the same performance as online AMC with nearest neighbor classifiers, but with drastically lower complexity. Simulations also provide a prescription for designing future SVM-based online fast AMC algorithms by appropriate selection of the training set size, kernel function, and feature set.

Finally, I demonstrate the ability of online AMC through supervised learning to optimize performance in nonlinear channels with non-Gaussian additive noise. Simulations show that online AMC through supervised learning is able to capture these impairments without explicit modeling. Consequently, I show that online supervised learning can tune AMC to the unique char-

Table 4.1: Important chapter specific notation.

$\lvert\mathcal{I}\rvert$	number of distinct transmission parameters sets
W	training set size
i	index for transmission parameter set (MCS)
$\mathbf{z}_w$	feature set $w \in \{0, 1, \ldots, W-1\}$ in training set
$\mathbf{q}$	feature set query for transmission parameter prediction
p	feature set dimension
h	inference function in two-stage classification
g	discriminant function in two-stage classification
$\mathcal{R}_i$	physical layer data rate for transmission parameter set i
F	frame error rate constraint
$d(\mathbf{z}_w, \mathbf{q})$	distance metric between training data entry and query

acteristics of each transmit/receive device pair without excessive analysis of performance in terms of device impairments. Moreover, I show that the degree of link adaptation performance loss with offline learning in channels impaired by non-Gaussian additive noise depends on the frequency-selectivity. This suggests that optimization of power amplifier backoff for improved SNR is achievable through online AMC with supervised learning.

4.1.2 Important Chapter Notation

In addition to the notation presented in the system model of Section 2.3, this chapter uses the notation in Table 4.1.

4.2 Classification for Online AMC

In this section I develop the procedure to complete AMC with online supervised learning through classifiers. To capture training data in real-time,

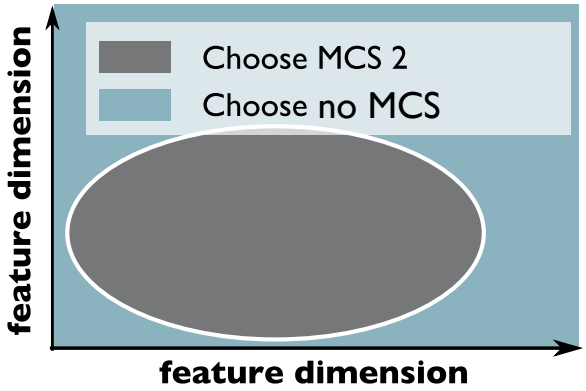


Figure 4.1: One-versus-none classification for MCS_2 with online AMC.

online AMC must update the training set as (un)successful frames are observed at the receiver.

4.2.1 One-versus-None Classification for each MCS

To update AMC classifiers on a per-frame basis, training data can only be updated with single-MCS information since it is not straightforward to infer the performance of one MCS from another. *I propose a new multi-class procedure that uses one-versus-none binary classifiers as the fundamental component of classification for AMC.* The newly proposed one-versus-none classifiers, illustrated in Fig. 4.2.1, choose between one class, i.e. MCS_i, and no classes, i.e. transmission is not advised. Equivalently I have created a new class which represents a null MCS.

Thus, for online AMC I must have $|\mathcal{J}|$ separate classifiers corresponding to transmission with each MCS. The training data for each classifier is composed of the feature set realizations extracted from the wireless channel during

frame transmission and the packet transmission result, i.e. failure or success. When a frame encoded with MCS_i is communicated successfully (success), this reinforces the classifier decision to use MCS_i, and when a frame is decoded in error (failure), this reinforces the classifier to advise the null MCS. Hence, if I execute one-vs-none classification for each MCS, I receive a set of candidate MCSs that are more likely to transmit frames successfully than not.

4.2.2 Addressing the FER Constraint

One-versus-none classification does not explicitly address the FER constraint. With one-versus-none classification, there are two classes for each MCS: success or failure. Therefore, given a feature set query $\mathbf{q}$, FER for MCS_i can be defined

$$\text{FER}_i(\cdot) \approx \Pr\left[\mathsf{failure}_i|\mathbf{q}\right] \tag{4.1}$$

which is the failure class posterior probability for MCS_i. Hence, to address the FER constraint with one-versus-none binary classification and training data captured in real-time, the class posterior probability must be available to the supervised learning classification algorithms.

4.2.3 Online AMC Classification Procedure Summary

The base procedure for online AMC to classify feature set realization $\mathbf{q}$ with training data captured in real-time is as follows (estimate of (4.1) denoted $\widehat{\text{FER}_i}$).

1. Search to find $\hat{i} = \arg\max_{i \in \mathcal{I}} \{\mathcal{R}_i : \widehat{\text{FER}_i} \leq \mathsf{F}\}$

2. If $\left|\left\{\hat{i}\right\}\right| > 1$ or $\hat{i} = \emptyset$, $\hat{i} \leftarrow \arg\min_{i \in \hat{i}} \widehat{\mathrm{FER}}_i$.

3. If $\hat{i}$ is still not unique, randomly choose single MCS.

Note that steps 2-3 determine a unique MCS for each feature set realization. Next, I will design an online AMC algorithm that exploits training captured in real-time.

4.3 Online AMC through Database Learning

In this section I propose the baseline online supervised learning AMC algorithm through nearest neighbor classifiers on databases of saved training data. Nearest neighbor provides a flexible framework for class posterior probability estimation. Moreover, databases allow for simple data refreshment and for preservation of diversity in the channels observed. The work in this section will be used to validate the performance of further development for online AMC.

4.3.1 Class Databases

Databases store training data and allow supervised learning algorithms to predict the best MCS. I am restricted to information that may be extracted from each received frame. The list of available, relevant information and its purpose is enumerated as follows.

- Successful Transmission: The CRC in the MAC allows the receiver to determine whether the packet was successfully transmitted or whether a

failure occurred.

- TX Node ID: The MAC address, or an equivalent unique identifier of each node, is available in the MAC header of each transmitted packet.[1] This information is important since several circuit impairments such as phase noise and power amplifier nonlinearity are linked to the transmitter hardware.

- N_f: In many wireless networks, the number of bits in each frame is not fixed. FER is a function of frame length.

- MCS_i: The class index i indicates the transmission parameters for each frame (e.g. symbol modulation order, error control coding type, sampling rate, etc.) which are known at the receiver to enable packet decoding.

- Channel Measurements: Wireless channel measurements are available at the receiver. This includes the linear time-invariant impulse response estimate of the channel and the noise variance estimate of additive noise at the receiver. It may also include measurements to test for non-Gaussian noise, nonlinearities in the channel, and interference.

[1] If a packet is not successfully received, the TX Node ID associated with this packet may not be correctly decoded. Fortunately, there are typically a large number of potential TX Node IDs such as with the 48-bit MAC address, so the probability of confusing MAC addresses is exceedingly small. It may occur, however, that feature set information from a significant number of erroneous packets is not entered into a database since a valid TX Node ID is not decoded. Empirically, I observe this when a gross error was made in the chosen MCS. Gross errors only occur, however, when the databases are not properly initialized. Later, I will discuss methods for database initialization which should remedy this problem to a large extent.

- Time of Packet Transmission: A local counter (clock) at the receiver can associate database entries with a time quantity. This timestamp is used to determine the age of database entries.

The aforementioned information may be exploited by supervised learning algorithms. This is accomplished through p-dimensional feature sets. Effectively, feature extraction is the compression of this list of relevant information to vectors in $\mathbb{R}^p$. Since I have treated the feature set in a general way, I cannot assume that the mapping between feature set realizations and the correct MCS does not change when transmitter hardware changes or for varying packet length. For example, if the feature set only represents the linear time-invariant impulse response of the channel, this does not account for nonlinearities in the transmitter or the effect of different packet length. Consequently, the database is divided by MAC address and further subdivided into different packet length regions as shown in Fig. 4.2. As a special case, a feature set that incorporates all factors that characterize the wireless channel between different transmitters and different packet lengths do not need this subdivision. To provide a general database structure, however, this subdivision remains.

As packets are decoded at the receiver, feature sets are created from channel measurements and stored in the database. The quality of the channel is represented by whether or not the frame was successfully decoded (CRC passed). Each database entry therefore includes the feature set, a timestamp to determine the age of the feature set measurement, and an index with which

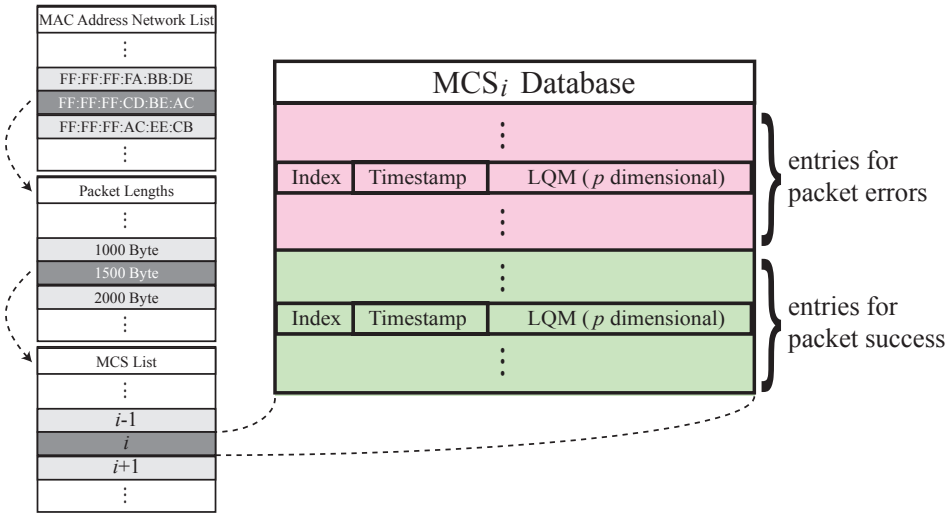


Figure 4.2: Class i database with features and timestamp for successful and unsuccessful transmitted frames

to reference the database entry. Each database entry corresponds to a single frame transmission and a single MCS. Therefore, feature set measurements can only reference a single MCS. As represented in Fig. 4.2, after the MAC address and frame length divisions the database is further subdivided into each MCS.

Databases, due to memory and search complexity constraints, are finite sized. Instead of constraining the total number of feature sets saved per MCS, it is important to reserve space for feature sets where the packets were successfully decoded and packets that were not successfully decoded. This will be discussed in more detail in Section 4.3.2. For each MCS I reserve N_{ack} spaces for feature sets corresponding to successfully decoded packets and N_{err} spaces for feature sets corresponding to packet errors.

4.3.2 Nearest Neighbor FER Estimation

To enable posterior probability estimation of class $\mathsf{failure}_i$ for all $i \in \mathcal{I}$, I modify the k-nearest neighbor (k-NN) classification algorithm to produce posterior probability outputs. The choice of k-NN is the result of two distinct advantages: (1) k-NN is non-parametric and no assumption is made on the functional mapping between feature realizations and the predicted physical layer parameter set (2) k-NN allows simple estimation of posterior probability outputs through the neighbor discovery process. The k-NN inspired link adaptation algorithm that exploits the databases is defined as follows.

AMC Procedure: The database for every MCS for the desired frame length is searched to find the k-nearest neighbors (in arbitrary distance metric $d(\cdot, \cdot)$) of $\mathbf{q}$. The database additionally shows how many of the k-nearest neighbors are associated with ACKs or NACKs. Let e_i, where $0 \leq e_i \leq k$, denote the number of nearest feature set neighbors associated with frame errors. Therefore, the maximum likelihood estimate of the posterior probability for class $\mathsf{failure}_i$ is

$$\widehat{\mathrm{FER}}_i = \frac{e_i}{k}. \tag{4.2}$$

Note that the number of databases that require a search is determined by $|\mathcal{I}|$. Consequently, $|\mathcal{I}|$ FER estimates are created. The selected MCS, $\hat{i} \in \mathcal{I}$, is the parameter set with maximum rate and estimated FER less than or equal to the FER upper bound F. This algorithm is summarized in Fig. 4.3.

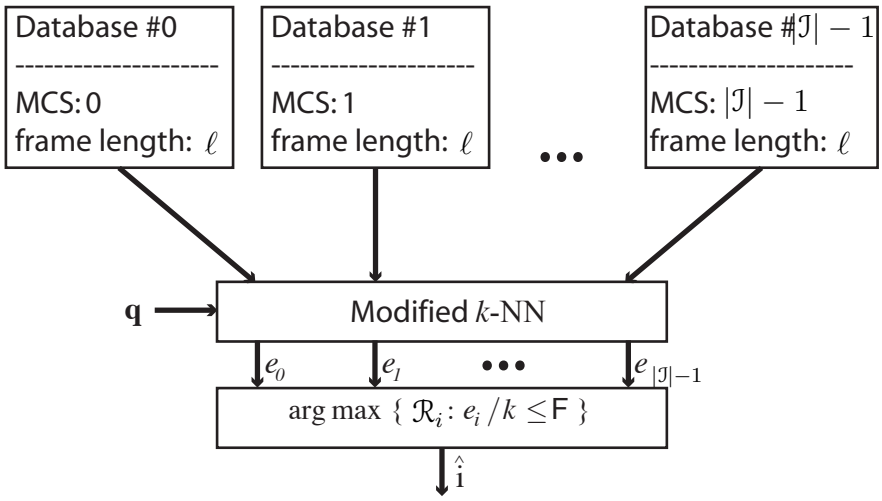


Figure 4.3: Link adaptation procedure to exploit databases through modified k-NN. $\hat{i}$ represents the index of the chosen MCS.

4.3.3 Database Initialization

In Section 4.3.2 when I defined the machine learning procedure, I assumed the MCS databases in Fig. 4.2 already contained information on past transmitted frames. In this section I consider the initialization of databases before the operation of the wireless link. Two approaches towards database initialization are presented: offline simulation/measurements and boundary values.

Offline Simulation/Measurements: This simple database initialization method exploits offline computer simulation of wireless channels or measurements to create entries for each MCS. This will fill the database with values that are roughly representative of the performance of the system. The

timestamp for simulation/measurement-based database entries should be set to the oldest value possible such that when the database is updated through the learning process in actual network operation it is able to replace these measurements first.

Boundary Values: Each database is loaded with boundary values on system performance. For MCS_i with designated TX Node ID and packet length, the N_{err} entries reserved for packet errors in each database are set equal to a feature realization that, under any circumstances, will result in packet failure with high probability. Likewise, the N_{ack} entries reserved for packet success are set equal to a feature realization that always results in a reliability that far exceeds the FER contraint (i.e. $\text{FER} \ll \mathsf{F}$). For the same reason as with offline simulations, boundary values in the database are associated with timestamps equal to the oldest value possible.

The advantage of using boundary values is that it allows the database learning to quickly respond to new observations when the wireless network is first activated, especially in fixed channels. Because boundary values of the feature space are far from practical measurements (in terms of the distance metric), new realizations of the channel are closer in $d(\cdot, \cdot)$ than the initialization values in the database. This is especially true for age-based updating (to be discussed in the sequel).

There are, however, two issues of warning when using boundary value filling in databases. First, the relationship between each feature set dimension and reliability (FER) must be monotonic, otherwise feature set distance

from a boundary value does not necessarily indicate its degree of reliability for each MCS. Second, the boundary values for the N_{ack} feature entries associated with packet success must be closer in $d(\cdot, \cdot)$ to typical feature set realizations than the N_{err} boundary values for feature entries associated with packet failure, otherwise online AMC with databases may initially result in conservative adaptation.

4.3.4 Database Maintenance

In the training stage of database learning, feature set measurements from each received data packet are stored in a database. Because databases are finite-sized, a database update procedure is needed when the storage capacity of a database is exceeded. In general, since I use a database initialization procedure, databases are always at the storage capacity. As a consequence, database updating is essential to the operation of online AMC with databases.

Age-Based Database Updating: The simplest and most straightforward operation of database updating takes an age-based approach. In age-based updating when a new feature set measurement is extracted from a received data packet it replaces the oldest value in the database. This approach is sufficient if the wireless channel is independent and identically distributed for each new packet arrival.

In practice, the wireless channel may be essentially static for many consecutive packets. Moreover, the statistical distribution from which realizations of the wireless channel are sampled may change over time. Consequently an

age-based database updating procedure does not preserve diversity in the feature space. This motivates the development of a database update procedure where an upper bound on the feature set density is created.

Density-Bounded Database Updating: The density of D feature sets $\{\mathbf{z}_l\}_{l=1}^{D}$ relative to a reference feature set $\mathbf{z}_0 = [z_{0,1}, z_{0,2}, \ldots, z_{0,p}]^T$ is defined equal to the quantity

$$\begin{aligned} \rho(\mathbf{z}_0, \{\mathbf{z}_l\}_{l=1}^{D}) &\triangleq \frac{D}{\left(\max_{l\in\{1,2,\ldots,D\}} \left\{\sum_{m=1}^{p} |z_{l,m} - z_{0,m}|^2\right\}\right)^{p/2}} \\ &= \frac{D}{r_{max}^{p}} \end{aligned} \tag{4.3}$$

where $r_{max} \triangleq \sqrt{\max_{l\in\{1,2,\ldots,D\}} \left\{\sum_{m=1}^{p} |z_{l,m} - z_{0,m}|^2\right\}}$. Density represents the number of feature set realizations in the database per unit volume. Scalar constants (in terms of π) used to precisely define volume in a hypersphere have been removed for simplicity. To preserve diversity in the feature space and to ensure a wide variety of channel realizations are maintained in the database, an upper bound $\overline{\rho}$ may be placed on the density.

Density bounding is achieved through the k-nearest neighbor algorithm. For each packet that is decoded in the receiver, there is an associated feature set $\mathbf{z}$ that is used to represent the state of the wireless channel. Hence, the database with N_{ack} or N_{err} elements (depending on whether the CRC passed) for a specified packet length and TX Node ID is searched to find the k-nearest neighbors $\{\mathbf{z}_l\}_{l=1}^{k}$. If $\rho(\mathbf{z}, \{\mathbf{z}_l\}_{l=1}^{k}) \leq \overline{\rho}$, the database is updated using the age-based approach. If $\rho(\mathbf{z}, \{\mathbf{z}_l\}_{l=1}^{k}) > \overline{\rho}$, however, the density threshold is exceeded. This is illustrated in Fig. 4.4.

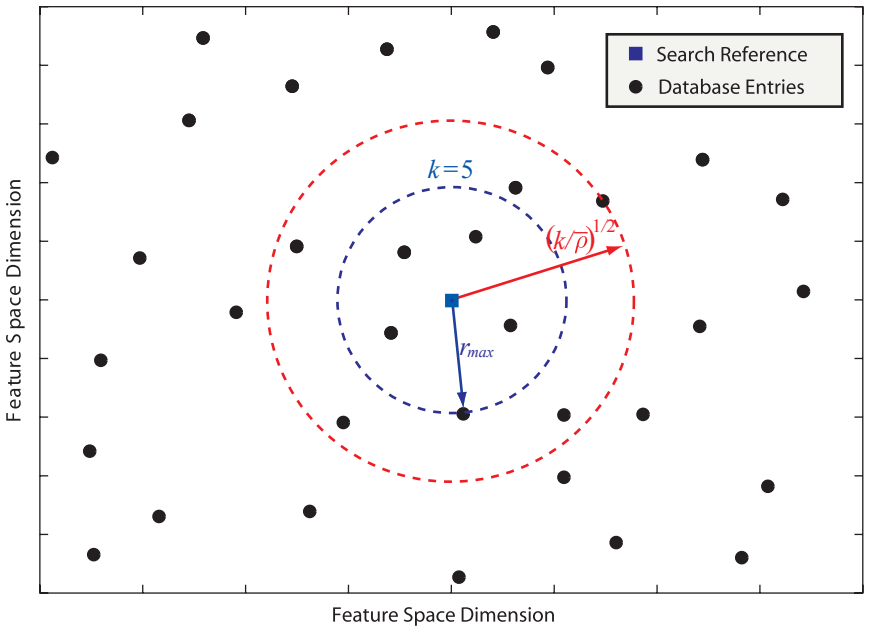


Figure 4.4: Illustration of feature diversity preservation through density bounding. In this 2-dimensional feature space (p=2) the k-NN search ($k = 5$) reveals a density measurement that exceeds $\overline{\rho}$ since $5/r_{max}^2 > \overline{\rho} \Leftrightarrow (5/\overline{\rho})^{1/2} > r_{max} \Leftrightarrow 5 > r_{max}^2\overline{\rho}$ with respect to a search reference.

When the density is exceeded with respect to $\mathbf{z}$, the update procedure does not replace the oldest entry in the database globally. Instead, $\mathbf{z}$ replaces the oldest feature set in the k-NN. In other words, the database entry with the oldest timestamp corresponding to one of the feature sets in the sequence $\{\mathbf{z}_l\}_{l=1}^{k}$ is replaced with $\mathbf{z}$. With the density-bounded database update procedure, the richness of feature sets in a single MCS database is preserved. Moreover, database learning is more agile since long runs of static channels will not pollute the database.

The density-bounding database update procedure just defined in this section implicitly assumes that all dimensions of the feature space are equally weighted (i.e. a Euclidean distance metric is used in the modified k-NN). It may not be advisable, however, to equally weight each feature space dimension if the statistical distribution of each feature space dimension extracted from the wireless channel is different than the other dimensions. Further, assuming the statistics of each feature space dimension align, the information value of each dimension may not be equivalent (i.e. a change in each dimension does not reflect equal change in rate adaptation). Consequently, I may redefine the density as a function of any arbitrary distance metric $d(\cdot,\cdot)$ such that

$$\rho(\mathbf{z}_0, \{\mathbf{z}_l\}_{l=1}^{N}, d) \quad \stackrel{\Delta}{=} \frac{D}{d_{max}^{p}} \tag{4.4}$$

where $d_{max} \stackrel{\Delta}{=} \max_{l\in\{1,2,\ldots,N\}} \{d(\mathbf{z}_0, \mathbf{z}_\ell)\}$.

4.3.5 Exploration

Although feature set density bounding is important to maintain richness in the database, especially during long runs of packets with static wireless channels, it does not necessarily maximize the performance of the rate adaptation procedure described in Section 4.3.2. If the database does not contain diverse or recent feature set realizations on all MCSs, AMC using k-NN may become conservative. To further explain this scenario, consider an example as follows.

> The physical layer only supports two rates. Hence, there are only two MCS: $\mathcal{I} = \{1, 2\}$. I arbitrarily assume that the data rate of class 2 exceeds the rate of class 1 ($\mathcal{R}_2 > \mathcal{R}_1$). For the current wireless channel and associated feature set realization $\mathbf{z}$, both rates can support reliability below the FER threshold F. Naturally, the learning algorithm should return $\hat{i} = 2$. The database for class 2, however, may not include sufficient past information on feature sets in close proximity to $\mathbf{z}$ or the database may contain past information that suffered impairments not captured in the feature set. Interference, for example, may have caused degraded performance that was not measured when the feature set was extracted from the wireless channel. As a result, $\widehat{\mathrm{FER}}_2 > \mathsf{F}$, when in reality the wireless channel supports FER statistics and class 1 is conservatively selected by the learning algorithm. Because the learning algorithm selects class 1 repeatedly, the outdated information in the database for class 2 is never flushed and the conservative rate selection problem persists.

To reduce conservative rate selection behavior it is necessary to *explore* the different MCS options. When class exploration is enabled, online AMC algorithms will assume that the MCS selected by the modified k-NN procedure, $\hat{i}$, is conservative, and completes a random mapping, E, to a MCS with a higher rate than $\hat{i}$. Thus $E(i) \in \mathcal{E}(i)$ where $\mathcal{E}(i)$ is the set of all MCSs such that $i' \in \mathcal{E}(i) \Rightarrow \mathcal{R}_{i'} > \mathcal{R}_i$. In many wireless networks, I can reduce the dimensionality of $\mathcal{E}(i)$ as a result of the natural ordering of MCSs in terms of reliability.[2] The functionality of the MCS exploration procedure is summarized in Fig. 4.5.

MCS exploration, while necessary to keep databases full of relevant information and maximize the throughput in rate adaptive systems, also leads to significant overhead since the reliability of traffic may be compromised during exploration sessions. Therefore, exploration should only be performed if the network condition permits. There are two primary network conditions for MCS exploration: during low-utilization intervals of the wireless network *and* when the FER drops below a lower threshold F_{lb}.

Lower Bounding the FER: The reliability of most wireless links is inversely related to the data rate of the MCS selected [48]. Hence, if the FER is well below the upper bound F, this is an indicator that the MCS being used is too conservative and MCS exploration should be initiated. To determine and prevent these conservative communication circumstances, a lower bound, F_{lb}, is placed on the measured FER. Therefore, during wireless network operation

[2]For a given $\text{MCS}_{i'}$ there may exist $\text{MCS}_{i''}$ such that $\text{FER}_{i''} > \text{FER}_{i'}$ for all feature set realizations. In the case that $i', i'' \in \mathcal{E}(i)$, I shall remove i'' from $\mathcal{E}(i)$.

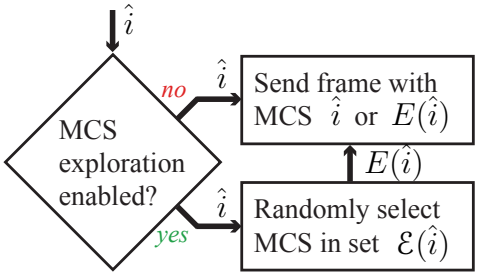


Figure 4.5: MCS $\hat{i}$ is returned by the modified k-NN algorithm in Fig. 4.3. If MCS exploration is enabled, frames will be transmitted using $E(\hat{i})$, i.e. randomly selected from the set $\mathcal{E}(\hat{i})$.

I desire to maintain $\mathsf{F}_{lb} \leq \text{FER} \leq \mathsf{F}$.

Low-Utilization of the Wireless Network: In many wireless networks, data traffic is bursty. There are often segments of time where the network is heavily used followed by segments of time when the network is sparsely used. For example, office WLANs experience high utilization during business hours with low utilization outside of business hours, during holidays, during lunch hour, during meetings, etc. Online learning may take advantage of this downtime, similar to the manner in which antivirus software takes advantage of low CPU utilization periods, to explore the MCS list for different packet lengths.

To fully automate MCS exploration, low utilization periods must be detected. If there are multiple nodes in the network also taking advantage of low utilization periods, the nodes must all be wary of generating too much traffic dedicated to exploration. Although the design of a protocol to perform

exploration in low-utilization periods is too detailed to be presented within the scope of this chapter, guidelines for such a protocol are listed as follows.

- The transmitter determines when to initiate exploration based on the contention observed in the wireless network.
- If priority or quality of service (QoS) is available for wireless packets, exploration packets are given the lowest priority.
- Exploration should only be available when data buffers are empty in the transmitter.
- Exploration should not cause latency problems or reduce the expected throughput of any data transfer over the wireless network.

Packet Snooping: For each packet transmitted over the wireless channel, I have assumed the receiver in the link is the only possible destination. Other nodes within the transmission range that are not the desired receiver of the wireless packet, however, may also listen to or *snoop* these transmissions. Moreover, snooping nodes may extract feature sets associated with the wireless channel between transmitter and the snooping node. Consequently, packet snooping provides free database information for nodes even when they are not participants in wireless exchanges.

Database Sharing with Homogeneous Links: Collectively, all the nodes in a wireless network have a wealth of information with regard to feature set realizations for different packet lengths and different MCSes. In gen-

eral, this information may not be shared because some factors that determine link reliability, such as amplifier nonlinearities and co-channel interference, are properties of the link that are not necessarily captured in feature sets. However, if nodes in a wireless network observe homogeneous links, defined as follows, database sharing is feasible.

Database learning is present in wireless networks, the links between arbitrary transmit/receive combinations are considered comparable if the feature set extracted from the wireless channel captures all factors that determine reliability. Consequently, the feature set must parametrize, but is not limited to, the packet length, interference effects, circuit nonlinearities, and non-Gaussian additive noise. If each link in comparable, this is a system with homogeneous links.

If the feature sets capture all the factors that determine performance, the mapping between feature set realizations and link reliability for each MCS will no longer depend on the time the feature sets were extracted. Consequently, it is possible to initially create a database that does not need to be updated during the operation of the network. Unfortunately, it is unlikely that any feature set will capture all factors that impact link reliability. Moreover, such a feature set will contain more information, increasing its dimensionality. Therefore, the curse of dimensionality may require exceedingly large training sets to complete rate adaptation through supervised learning.

4.3.6 Experimental Results

All experiments in this section are conducted on Hydra, an IEEE 802.11n standard multiple-antenna wireless network testbed created at the University of Texas at Austin. Hydra transceivers use general purpose processors (notebook computers) to complete the processing and state handling for all layers of the communication system. The radio frequency (RF) circuits, which operate at the 2.45 GHz frequency, are attached to the general purpose processor through a USB 2.0 connection. Due to the flexibility of this software-defined radio implementation it is relatively straightforward to implement online AMC through database learning on Hydra. For more details, the reader is encouraged to reference earlier publications on experimentation with Hydra [21, 62].

Several database learning parameters are configured for the implementation on Hydra. The feature set extracted from the wireless channel state uses ordered post-processing SNR developed in Chapter 3 with dimension $p = 4$. The k-NN search uses the Euclidean distance metric and the information databases are allocated 1000 entries for ACKs and NACKs ($N_{ack} = N_{err} = 1000$) in each database for parameter set $i \in \{1, \ldots, |\mathcal{I}|\}$. The number of MCSs, $|\mathcal{I}| = 16$, correspond to the 16 distinct transmission modes (modulating and coding schemes) of IEEE 802.11n systems with 2 transmit and 2 receive antennas (as implemented in Hydra). The target physical layer FER is 10% ($\mathsf{F} = 0.1$), since it often indicates a threshold for limiting overhead caused by retransmissions in higher layers [13]. To enable observable

$\widehat{\text{FER}} \leq 0.1 \Rightarrow k \geq 10$. I have chosen $k = 10 \Rightarrow$ one observable error is allowed in the 10 nearest neighbors before $\widehat{\text{FER}}$ violates the FER threshold.

The experimental results in this section demonstrate the utility of online AMC through database learning. To that end I first compare the performance of database learning with baseline online link adaptation algorithms. The performance advantage of database learning (with FER lower bounding ($\mathsf{F}_{lb} = 0.03$) and feature set density bounding ($\overline{\rho} = 10$)) over the baseline algorithms will be illustrated with Hydra experiments in static and dynamic wireless channels. The second set of experiments use emulated wireless channels to demonstrate the performance characteristics of density bounding feature sets and exploration.

Performance in Static and Dynamic Channels: To demonstrate the performance advantage that online AMC through database learning offers, I choose baseline realizations of online and offline link adaptation. The baseline autonomous rate adaptation algorithm is auto-rate fallback (ARF), which uses ACK/NACK information to increase or decrease the data rate. In ARF, when 10 consecutive ACK messages are received, the MCS with the next higher data rate is chosen. If 2 consecutive NACK messages are received, the next lower rate is chosen. For offline link adaptation, I have chosen to use a k-NN supervised learning classifier with 4-dimensional subcarrier-ordered SNR feature sets, since this has was shown to yield superior performance in Chapter 3.

The wireless communication experiments are conducted between two

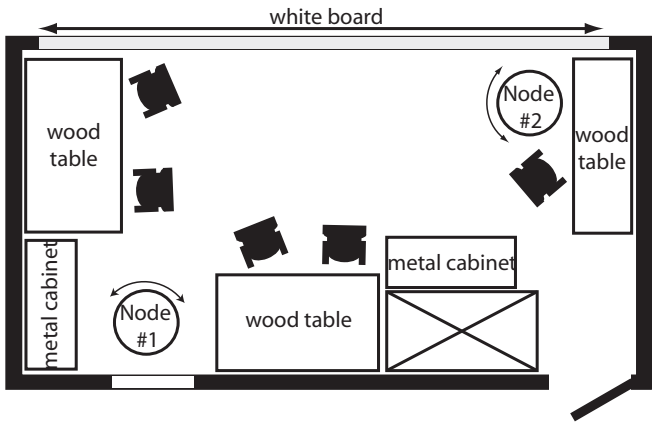


Figure 4.6: Laboratory experiment environment. Node #1 and Node #2 each contain Hydra IEEE 802.11n transceivers with online AMC through database learning. Communication is bidirectional between each node. For static channels, the nodes are not moved. In dynamic channel measurements the nodes are placed on oscillating platforms as indicated by the double arrows.

transmitting/receiving nodes in an office environment as illustrated in Fig. 4.6. This environment was chosen for a number of reasons. First, this represents a typical usage scenario for wireless networks such as laptop to internet base-station communication or wireless data transfer from mobile devices to computers (e.g. Bluetooth, wireless USB). Second, the multiplicity of objects in the office environment create interesting wireless channels through electromagnetic wave reflections off walls, tables, metal cabinets and chairs. This increases the challenge of rate adaptation. Finally, because this office is dedicated to the students running the experiments, it can be easily controlled. For static channel experiments, where all moving objects must be removed from the room, this is especially important.

In static wireless channels link adaptation attempts to determine the

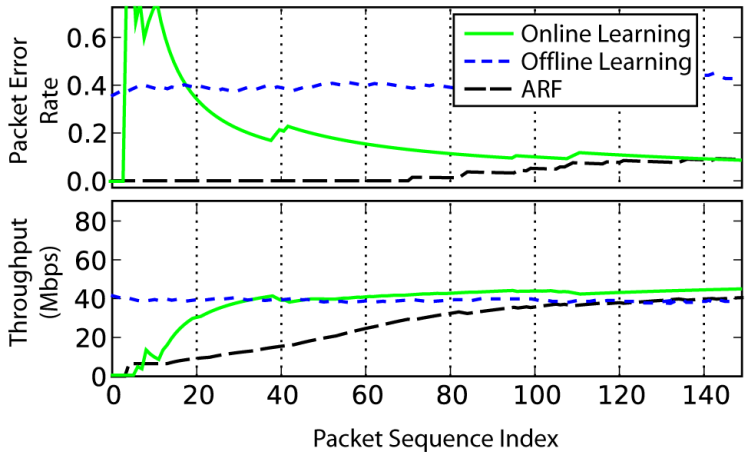


Figure 4.7: FER and throughput for 150 packets transmitted on 1 sec intervals from Node #1 to Node #2 in a static channel. Baseline offline and online link adaptation strategies are supervised learning with 4-dimensional subcarrier-ordered SNR feature sets and ARF, respectively. Online AMC databases are initialized with boundary values.

MCS that maximizes the data rate while satisfying the reliability constraint (FER$\leq$ F). For offline link adaptation, if machine learning algorithms are trained incorrectly, they will always perform poorly. This is seen in Fig. 4.7 where I compare the packet error rate and throughput of offline supervised learning, ARF, and online database learning in static wireless channels. Because the offline learning AMC algorithm was trained using emulated channels, it is unable to capture all the effects of a real wireless environment that do not follow the traditional models of wireless communication. This results in the violation of reliability constraints and reduced effective throughput. Offline link adaptation may improve performance by being trained/calibrated through real channel measurements. Unfortunately, it is difficult to observe

all the potential channel scenarios necessary to train the learning algorithms. Moreover, the model deviations are unique to each wireless device and may evolve over time. By comparison, ARF, the baseline autonomous rate adaptation procedure, eventually selects the correct transmission mode to maximize throughput. ARF, however, suffers from slow convergence since it does not observe explicit channel information.

In wireless channels that change quickly, the dynamic channel conditions cause ARF to conservatively select transmission modes since ARF is limited by the worst channel states that cause consecutive NACKs. This is demonstrated in Fig. 4.7 where the throughput of ARF is significantly less than the throughput of online AMC with database learning. The FER of ARF is well below the target, which also suggests that ARF is conservatively adapting rate. For both static and dynamic channels, online AMC is clearly preferred. Online AMC with database learning performance converges more quickly than ARF, but slower than dependent rate adaptation. This is due to the databases starting with only boundary value information. Once the databases have sufficient history of wireless network performance, database learning does not suffer from convergence issues. In other words, convergence is a startup cost of database learning suggesting the performance improvement of database learning over baseline link adaptation strategies is even more drastic than Figures 4.7 and 4.8 display.

Feature Set Diversity Preservation and Exploration: To investigate the ability of database learning to maintain density upper bounds in

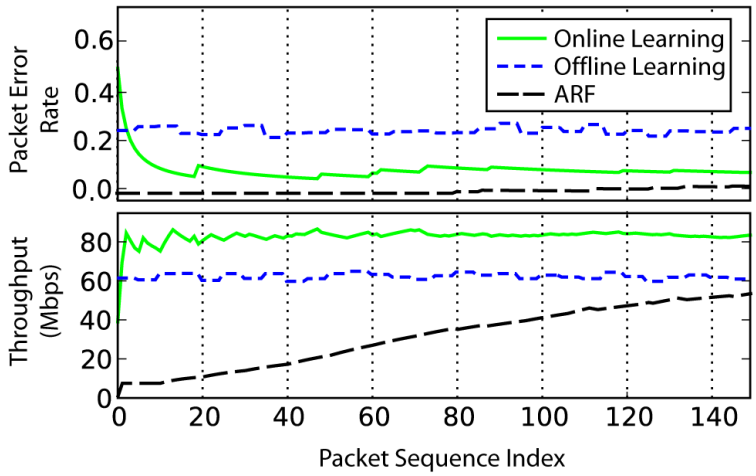


Figure 4.8: FER and throughput for 150 packets transmitted on 1 sec intervals from Node #1 to Node #2 with Node #2 oscillating over 120° with a period of 14.5 sec. Baseline offline and online link adaptation strategies are supervised learning with 4-dimensional subcarrier-ordered SNR feature sets and ARF, respectively. Online AMC databases are intialized with boundary values.

the feature space of information databases and reduce conservative rate selection, I run database learning with IEEE 802.11n in Hydra through two distinct channel distributions in a sequential fashion. Because it is difficult to control the statistical distribution of the channel model and reproduce this accurately in physical channels, I have chosen to use channel emulation and maintain control. The concern here is not to characterize the intricate details of a real wireless channel, but to demonstrate that, without intelligent feature set density bounding and exploration, conservative rate selection will occur.

The information databases are initialized using offline simulations since the emulated channels will not provide effects outside of these simulations. The offline simulations consist of 4×10^4 1 kB packets, transmitted with random

Table 4.2: Channel distribution and packet lengths for different stages of experiments that validate feature set density bounding and exploration

	Channel Distribution	**# Packets**	**Packet Length**
Initialization	3-tap, $10-35$ dB SNR	4×10^4	1 kB
Stage I	1-tap, $10-35$ dB SNR	2×10^4	1 kB
Stage II	3-tap, $10-35$ dB SNR	2×10^4	1 kB

transmission modes and independent channel realizations drawn from a 3-tap Gaussian channel distribution with a uniform power delay profile. The relative large-scale path loss is independently sampled for each packet uniformly over the interval $[10^{1.0}\ 10^{3.5}]$ (roughly providing $10-35$ dB SNR). The transmitted power of each packet and the variance of the complex Gaussian additive noise at every receive antenna are normalized to 1.

In Stage I, a frequency-flat channel is emulated for 2×10^4 1 kB packets. In Stage II, the same channel distribution used for database initialization provides for emulation of 2×10^4 1 kB packets. The different stages of the channel emulation are summarized in Table 4.2. After Stage I, without feature set density bounding, the databases will lose significant information about the 3-tap channels that was obtained during database initialization. This is demonstrated in Fig. 4.9 with varying feature set density upper bounds $\overline{\rho}$. The throughput and FER in Stage II show that database learning performs conservatively if feature set density bounding is not provided (i.e. $\overline{\rho} = \infty$). Moreover, if I choose too large of an upper bound ($\overline{\rho} = 100$), the databases are not preserving a sufficient amount of diversity and if the databases choose too small of an upper bound ($\overline{\rho} = 0.001$), density bounds will not be violated

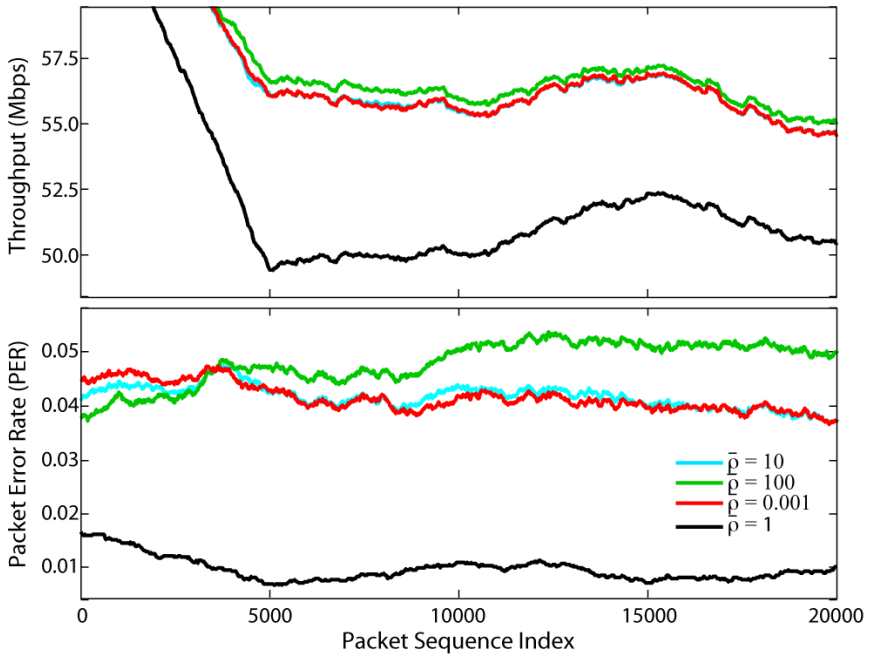


Figure 4.9: Throughput and FER in Stage II of feature set diversity preservation with varying density bounds ($\overline{\rho} = \{0.001, 10, 100\}$). If $\overline{\rho} = \infty$ feature set diversity preservation is turned off. The plots correspond to 2×10^4 1024 byte packets with independent channel realizations in 3-tap frequency selective channels where SNR is uniformly distributed between 15 and 35 dB. The plots use a sliding window of 5000 samples to measure throughput/FER.

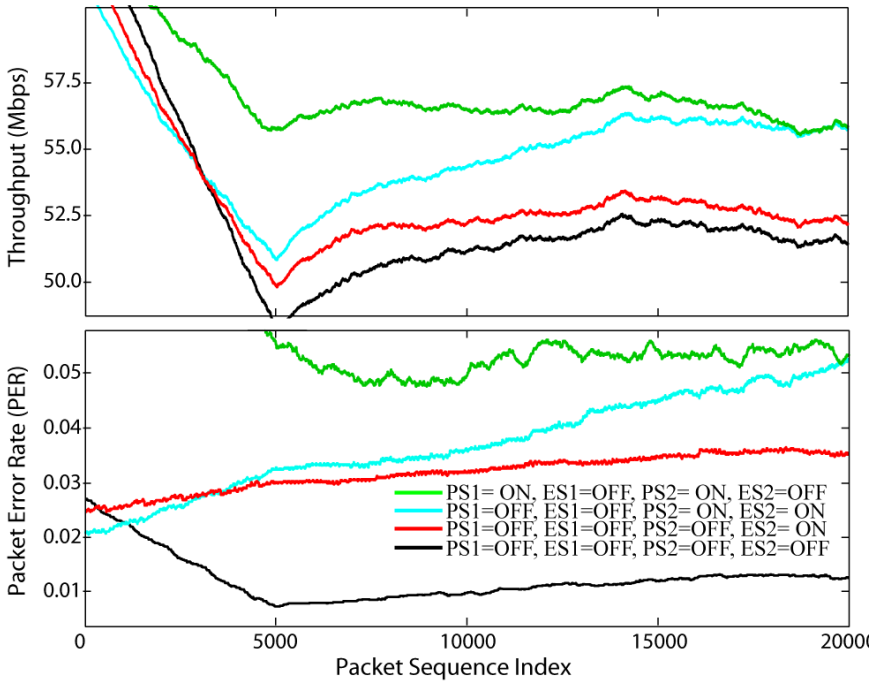


Figure 4.10: Throughput and FER in stage II to demonstrate the result of exploration with FER lower bounding ($\mathsf{F}_{lb} = 0.03$). Different combinations of exploration and feature set diversity preservation ($\overline{\rho} = 100$) are tested. PS1, PS2 refer to feature set diversity preservation in Stages I and II, respectively and ES1, ES2 refer to exploration with FER lower bounding in stages I and II, respectively. The plots use a sliding window of 5000 samples to measure PHY throughput ($(1 - \text{FER}) \times (\text{Average Data Rate})$) and FER.

enough since small variations in the channel and feature estimation will cause slight feature set differences.

Even if feature set density bounding is provided, conservative rate selection will still occur if insufficient or outdated information exists in the databases. In Fig. 4.10 I examine the effects of feature set density bounding and exploration through FER lower bounds. Fig. 4.10 shows that learning

in Stage II can help recover some of the lost diversity in the feature space after Stage I. Moreover, when exploration and feature density bounding are used together, the lost diversity can be completely recovered in a timely manner (2×10^4 packets of traffic can result from < 5 sec of traffic in IEEE 802.11n). Likewise, new statistical distributions or environmental factors can be discovered in a timely manner through exploration.

4.3.7 Complexity Discussion

In this section I consider the complexity of online AMC through database learning. From a wireless device design perspective, the memory dedicated to databases is a major disadvantage since a large quantity of reserved memory corresponds to valuable silicon space on single chip solutions [30]. Fortunately, as discussed in Section 4.3.7, memory space can be significantly reduced. Moreover, if database learning and its memory reserves are placed in firmware [102], it is feasible to use a host (e.g. a laptop/desktop computer host for an IEEE 802.11 NIC) for long-term storage of some database information. In Section 4.3.7 I present the search complexity of k-NN and discuss methods to reduce worst case search latency through database grid structures and approximations of k-NN.

Database Memory Requirements: Online AMC with database learning, strictly implemented, requires significant memory storage for each of the information databases. For example, for each receiver let L designate the number of distinct packet lengths used for wireless transmission, let M de-

termine the number of transmitters (MAC addresses) to track, let B_t denote the number of bytes per timestamp, and let B_f equal the number of bytes per feature set per dimension. Hence, I find

$$\text{total database memory} \geq ML|\mathcal{I}| \times (N_{err} + N_{ack}) \times (B_t + pB_f) \text{ bytes.} \quad (4.5)$$

In a typical network, the number of distinct packet lengths is limited by the source of traffic. Similarly, most wireless devices (besides base stations) do not need to maintain history for a large number of transmitters. Further, in the IEEE 802.11n testbed, I have empirically found $N_{err} = N_{ack} = 1 \times 10^4$ to be more than sufficient for nearly all operating conditions. Additionally, for a 2 transmit antenna, 2 receive antenna IEEE 802.11n link, $|\mathcal{I}|$=16. Therefore, if I assume 4 bytes for a timestamp (UTC time code) and 4 bytes for each feature dimension (floating point value), a typical IEEE 802.11n device using the ordered SNR feature sets from Section 4.3.6 might require $\approx$ 100-200 MB of database storage.

For current wireless devices, memory storage of this size will be difficult to accomodate, especially for on-chip solutions. Fortunately, there are several instances where memory storage can be compressed by orders of magnitude. For example, with the ordered SNR feature sets considered in the IEEE 802.11n prototype, there is only a finite range of potential values in each dimension due to constraints on feasible transmitted power and receiver sensitivity. Moreover, high numerical precision (floating point) is not needed for each dimension. UTC timestamps are also excessive since dynamic factors

observed in the wireless network (e.g. ambient interference, nonlinearities due to temperature change, etc.) change on the order of minutes and hours, not months or years. It is therefore possible to reduce the timestamp storage size significantly. Further, I have observed that reducing the number of feature sets by an order of magnitude (i.e. from 1×10^4 to 1×10^3 for N_{err}, N_{ack}) does not significantly reduce the performance of rate adaptation. Consequently, with further design, firmware implementations will provide the necessary memory to operate database learning.

$k-$NN Search Complexity: The most significant and unique processing requirement for online AMC with database learning is the k-NN search through a MCS database. For each n-sized database ($n = N_{ack} + N_{err}$) with p-dimensional entries, k-NN implemented with a brute force search requires complexity $O(pn)$ and therefore each database search displays complexity $O(|\mathfrak{I}|pn)$. Brute force search, however, does not exploit any structure in the information databases. A number of k-NN algorithms reduce the complexity of searches by restructuring databases into a grid, such that more intelligent searches can be completed. It is well known that grid structuring algorithms with $O(pn \log n)$ complexity to reassign the database memory can reduce the *expected* search complexity to $O(\log n)$ [29]. Therefore, the search complexity can be reduced by adding complexity to the database storage mechanism. For online AMC, database grid structuring is less latency critical since it can be performed periodically when packets are not being transmitted/received or in parallel to packet transmission/reception.

Unfortunately, although the expected complexity is favorable for k-NN with structured databases, the worst case search complexity does not follow $O(\log n)$ [3]. Moreover, the expected search complexity is only valid for a uniformly distributed feature space in each information database. In practice, I have empirically observed that practical wireless channels, even with database density bounding algorithms, tend to create databases with clustered feature spaces. That is, the n feature space realizations from wireless channels have regions of high and low density. Consequently, I desire k-NN search algorithms that can guarantee low-complexity in the worst case since I cannot make assumptions on the statistical distribution of entries in the feature space.

Another means of reducing complexity in the k-NN search is to accept an error tolerance ϵ in the nearest neighbor. Thus, given a query $\mathbf{q} \in \mathbb{R}^p$ and a true nearest neighbor $\mathbf{z}'$, an approximate nearest neighbor (ANN) $\mathbf{z}$ satisfies $d(\mathbf{q}, \mathbf{z}) - d(\mathbf{q}, \mathbf{z}') \leq \epsilon$. Using this principle I can complete the k-ANN search algorithm to guarantee search order complexity of $O((c+kp)\log n)$ with database structuring complexity of $O(pn \log n)$ for some $c \leq p\lceil 1 + 6p/\epsilon \rceil^p$ [3].

4.4 Online AMC with Support Vector Machines

In Section 4.3 I proposed a design for online AMC with database learning through nearest neighbor classifiers. While the performance is impressive, the complexity of the implementation limits its application in practice. In this section I propose an alternative online AMC procedure through support vector machines (SVMs). SVMs have received significant interest recently due

to their ability to discriminate intricate nonlinear class region boundaries in a similar manner to neural networks, but often alleviating the complexity, optimization, overfitting, and interpretation issues that have long been associated with neural networks. The dual representation of SVM classifier optimization is particularly valuable for online AMC, since it allows the classifier to smoothly evolve over time. Most importantly, as I show in this section, SVMs can drastically reduce the implementation complexity without loss in performance.

4.4.1 SVM Classifier

To enable simple and tractable algorithms, many classifiers (including support vector machines) are implemented through a two-stage functional mapping [6]. The first stage, the inference function, h, evaluates the structure of class boundaries to create distinct mappings from regions in the feature space to a real number. In the second stage a discriminant function, g, untangles the inference mapping and maps each point on the real line to a positive integer that indexes each class. Altogether, a two-stage classifier, $f(\mathbf{q}) = g(h(\mathbf{q}))$, completes the mapping

$$f : \mathbb{R}^p \rightarrow \mathbb{R} \rightarrow \{0, 1, \ldots, |\mathcal{I}| - 1\} \tag{4.6}$$

for feature set query $\mathbf{q}$.

SVMs, also known as maximum margin classifiers, are binary classifiers, meaning they place class label $y \in \{+1, -1\}$ on feature set realizations

according to distance from each respective margin. SVMs construct a margin, as illustrated in Fig. 4.11 for a two-dimensional feature set, that defines the class regions of the feature space, i.e, success/failure in the online AMC one-vs-none classification, through $+1/-1$ margins. For feature set realization $\mathbf{q}$, SVMs manifest the margin through inference function $h(\mathbf{q}) = \mathbf{w}^*\phi(\mathbf{q}) + b$ where $\mathbf{w}$ is the linear discriminant vector, $\phi(\mathbf{q})$ is the SVM feature transform[3], and b is the margin bias. The two class outcomes are provided by the discriminant function $g(h(\mathbf{q}))$ such that $(-\infty, 0) \mapsto -1$ and $[0, \infty) \mapsto +1$. Each of the $|\mathcal{I}|$ one-versus-none SVM classifiers are defined in terms of an W-sized training data set. I define the wth element of ith classifer's training data set set with $\mathbf{z}_{i,w}$ $\forall w \in \{0, 1, \ldots, W-1\}$. For the remainder of this section I drop the subscript i with the understanding that SVM optimization procedures must be completed for each of the $|\mathcal{I}|$ binary classifiers.

If C is the penalty for feature set realizations that require $\xi_w > 0$ in the non-separable case, maximum margin construction for SVMs is equivalent to finding the arguments that solve the quadratic optimization problem

$$\begin{aligned} \min_{\mathbf{w}\in\mathbb{R}^p, b\in\mathbb{R}, \boldsymbol{\xi}\in\mathbb{R}^W} \quad & \left\{ \tfrac{1}{2}\mathbf{w}^*\mathbf{w} + C\sum_{w=0}^{W-1} \xi_w \right\} \\ \text{constraining} \quad & y_w \left(\mathbf{w}^*\phi(\mathbf{z}_w) + b\right) \geq 1 - \xi_w \\ & \xi_w \geq 0 \end{aligned}$$

which maximizes the distance between margins. In online AMC, the one-

[3]In SVM literature $\phi(\mathbf{q})$ is often called the feature set. Here, I have assumed preprocessing before the SVM which defined $\mathbf{q}$ as the feature set. For clarity I will maintain the current definition of $\mathbf{q}$ as the *feature set* and $\phi(\mathbf{q})$ as the *SVM feature set* throughout.

versus-none classification is non-separable since feasible feature space realizations cannot provide deterministic mappings to class outcomes. Hence, C is a parameter that must be tuned to optimize classifier performance and, ultimately, minimize the probability of incorrectly selecting the desired MCS given aribitrary feature set realizations.

The kernel function is defined $\kappa(\mathbf{z}_w, \mathbf{z}_{w'}) := \phi(\mathbf{x}_w)^* \phi(\mathbf{z}_{w'})$ which represents the correlation of objects in the SVM feature space. If I define the matrix $\mathbf{Q} \in \mathbb{R}^{W \times W}$ where $[\mathbf{Q}]_{w,w'} := y_w y_{w'} \kappa(\mathbf{z}_w, \mathbf{z}_{w'})$, maximum margin construction may also be formulated through the Lagrangian dual and the KKT conditions

$$\begin{aligned} \min_{\boldsymbol{\alpha} \in \mathbb{R}^W} \quad & \left\{ \tfrac{1}{2} \boldsymbol{\alpha}^* \mathbf{Q} \boldsymbol{\alpha} - \mathbf{e}^* \boldsymbol{\alpha} \right\} \\ \text{constraining} \quad & \textstyle\sum_{w=0}^{W-1} y_w \alpha_w = 0 \\ & 0 \leq \alpha_w \leq C \end{aligned}$$

where $\mathbf{e}$ is a vector of all ones and the equivalent definition of the inference function becomes $g(\mathbf{q}) = \sum_{w=0}^{W-1} y_w \alpha_w \kappa(\mathbf{z}_w, \mathbf{q}) + b$. Kernel functions generalize the shape of the class region boundaries. Linear kernels are defined $\kappa_{\text{lin}}(\mathbf{z}_w, \mathbf{z}_{w'}) := \mathbf{z}_w^* \mathbf{z}_w$ and provide hyperplane boundaries between classes. Polynomial kernels are defined $\kappa_{\text{poly}}(\mathbf{z}_w, \mathbf{z}_{w'}) := (\mathbf{z}_w^* \mathbf{z}_w)^d$ for dimension d. Finally, the popular radial basis function kernel, $\kappa_{\text{rbf}} := \exp\left(-\beta \|\mathbf{z}_w - \mathbf{z}_{w'}\|^2\right)$ for $\beta > 0$ defines boundaries in terms of exponential scaled Euclidean distance from support vectors.

The dual formulation has an insightful interpretation in terms of the optimal arguments α_w for $w \in \{0, 1, \ldots, W-1\}$. Each α_w has three potential

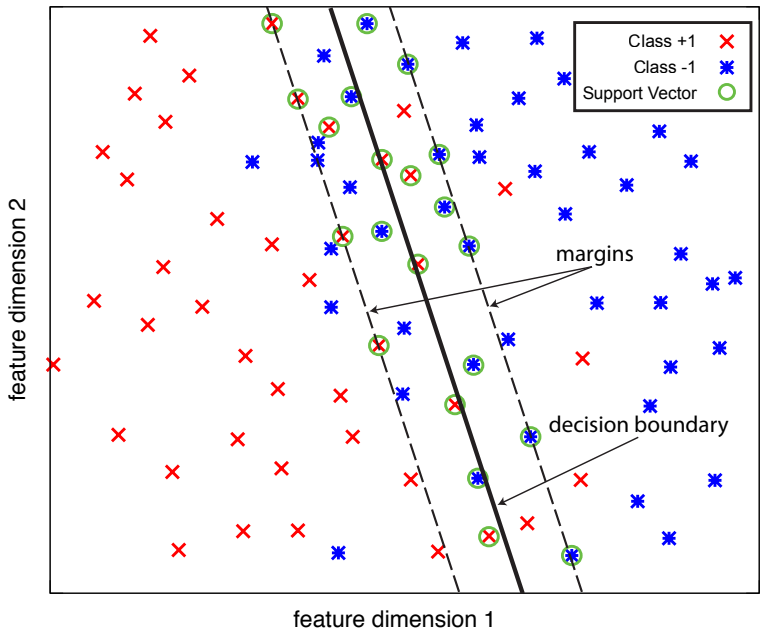


Figure 4.11: Illustration of margins and support vectors in training set feature space.

interpretations as follows and as illustrated in Fig. 4.11.

1. $\alpha_n = 0$: $\mathbf{z}_w$ does not contribute to the inference function.

2. $0 < \alpha_w < C$: $\mathbf{z}_w$ lies on the margin and is referred to as a *support vector* for class y_w.

3. $\alpha_w = C$: $\mathbf{z}_w$ is inside the margin (potentially misclassified) and is also a support vector for class y_w.

The term support vector is intuitive since they are the only elements of the training set that contribute to the inference function.

4.4.2 Online AMC Classification Procedure for SVMs

With nearest neighbor classifiers, the posterior probability class estimate was readily available through neighbor discovery. With SVMs the only useful information is the output of the inference function. Hence, I create a regression function, r, which completes the mapping $r : \mathbb{R} \rightarrow [0, 1]$ from the inference function output to the posterior probability.

Next, I provide the base algorithm for online AMC to classify feature set realization $\mathbf{q}$ with the two stage classifer framework given one-versus-none classifers and discriminant functions replaced with regression functions. I use the subscript i to identify each of the $|\mathcal{J}|$ separate regression and inference functions.

1. Compute h_i $\forall i$ from the training data.
2. Compute r_i $\forall i$ from all h_i and the training data.
3. Complete mapping $r_i(h_i(\mathbf{q}))$ $\forall i \in \mathcal{J}$.
4. Search to find $\hat{i} = \arg\max_i \{(1 - r_i(h_i(\mathbf{q})))\mathcal{R}_i : r_i(h_i(\mathbf{q})) \leq \mathsf{F}\}$
5. If $\left|\left\{\hat{i}\right\}\right| > 1$ or $\hat{i} = \emptyset$, $\hat{i} \leftarrow \arg\min_{i \in \hat{i}} r_i(h_i(\mathbf{q}))$.
6. If $\hat{i}$ is still not unique, randomly choose single MCS.

Note that steps 5-6 determine a unique MCS for each feature set realization. It is also notable to observe that $r_i(h_i(\mathbf{q}))$ provides an estimate of $\mathrm{FER}_i(\cdot)$. While steps 3-6 are necessary to predict the best MCS for each new channel

realization, steps 1-2 need not be completed for each new (un)successful frame observation at the receiver that enters the training set. For example, data can be collected per-frame, but the regression and inference functions can be evaluated with a lower periodicity. Note that again I drop the i subscript for notation simplicity.

4.4.3 Posterior Probability Prediction (FER Prediction)

In the proposed solution, I apply regression functions after the SVMs to complete online one-versus-none classification MCS i for each $i \in \{0, 1, \ldots, |\mathcal{I}|-1\}$. Assuming that the inference function outputs are normally distributed, the prior probabilities have the sigmoid form

$$\Pr\left[y = 1|h(\mathbf{q})\right] = \left(1 + \exp(A_1 h(\mathbf{q})^2 + A_2 h(\mathbf{q}) + A_3)\right)^{-1} \tag{4.7}$$

for constants A_1, A_2, and A_3. Because this function is non-monotonic for some realizations of constants A_i, $i \in \{1, 2, 3\}$ Platt suggested fitting inference outputs to the simpler sigmoid form

$$(1 + \exp(B_1 h(\mathbf{q}) + B_2))^{-1} \tag{4.8}$$

for constants B_1 and B_2 [76]. Studies have shown that the output of SVM inference functions are capable of providing accurate posterior class probabilities using the fitting function in (4.8) [49, 69]. Fitting in terms of the training set can be accomplished through many algoritihms including Levenberg-Marquardt. Here, I choose Newton's method with backtracking as prescribed in [57].

4.4.4 Evaluation of Online AMC in IEEE 802.11n with SVMs

I complete numerical simulations using the IEEE 802.11n standard [43]. The simulation parameters are summarized as follows.

- 20 MHz channels
- 2 receive antennas, 2 transmit antennas
- MCS_0 through MCS_{15} (1×1 and 2×2)
- 10% FER constraint ($\mathsf{F} = 0.1$)
- 128 byte packet length
- perfect RX synchronization and channel estimation
- zero-forcing equalization
- online AMC with proposed SVM (linear and RBF kernels) and database learning nearest neighbor classifiers ($k = 10$)
- ordered post-processing SNR feature sets

Although the algorithm is motivated for systems where nonlinearities and intractabilities limit the accuracy of optimization-based AMC algorithms, I initially proceed with linear channels generated from the IEEE 802.11n data set [19]. This data set provides 3.2×10^4 channel realizations to cover a range of over 30 dB in average SNR (assuming normalized transmit power and noise

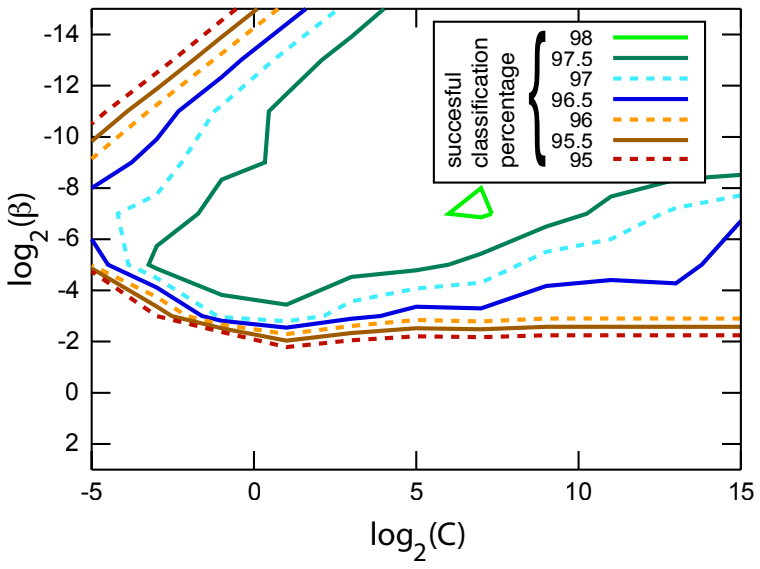


Figure 4.12: Sample cross validation contour plot for MCS_{15} in the training data set.

variance). For the evaluation I will use 4 tap Gaussian channels with uniform power delay profiles, resulting in severe frequency selectivity. The standard SVM algorithm is implemented with LibSVM [11]. All simulations are programmed in C++.

Cross-Validation: To optimize the performance of SVM classifiers, the penalty C and the kernel parameter β (for RBF kernels only) must be tuned. This is done through a grid cross-validation search with the entire training data set (i.e. 3.2×10^4 channel realizations and corresponding packet transmission results). Cross-validation is performed for all $|\mathcal{J}|$ online AMC SVM classifiers with both linear and RBF kernel functions. The contour plot in Fig. 4.12 illustrates the results of cross-validation for a single MCS. At the

conclusion of cross-validation it was determined that all $|\mathcal{J}|$ SVM classifiers are near optimal with $C = 10$ (linear and RBF kernels) and $\beta = 0.005$ (RBF only) and these settings were fixed for the remainder of simulations in this chapter. It should be noted that cross-validation merely finds parameter values that maximize classification probability, but does not necessarily minimize error in sigmoid regression for posterior probability prediction. Empirical results, however, have shown that cross-validation often gives good results for sigmoid regression as well.

Variable Training Length: It is critical that online AMC learns how to correctly select MCS in a timely manner since subpar adaptation performance results in lost throughput. Fig. 4.13 shows the average throughput (estimate of $\mathbb{E}_{\mathbf{q}}\left[\max_i\{(1-\mathrm{FER}_i(\cdot))\mathcal{R}_i : \mathrm{FER}_i(\cdot) \leq \mathsf{F}\}\right]$) as a function of the size of the training set, i.e. the number of packets observed over the channel for a single MCS. In general, the steady state performance is very similar for all classifiers. The throughput gap between a genie-aided AMC strategy and all simulated classifiers starts to level out at around 1000 channel/frame samples per MCS. Online AMC with SVM and linear kernels, however, learns faster. At $W = 60$ training samples per classifier, SVM with linear kernels is roughly the same as the performance with $W = 1000$ training samples. RBF kernels, however, require more training data, and this is reflected in the plot which shows that its throughput gradually increases with training size. Nearest neighbor trains the slowest, but its results are still comparable to online SVM.

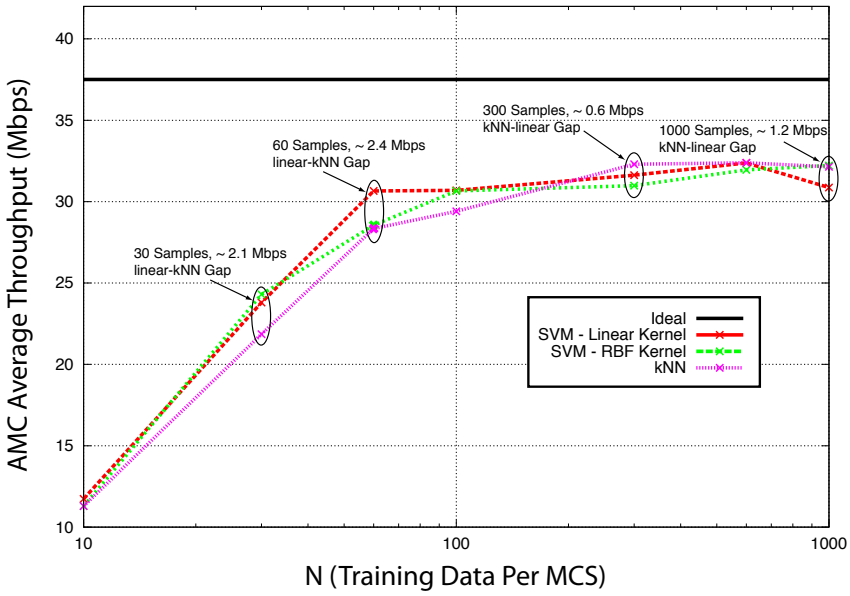


Figure 4.13: Throughput of online AMC averaged over 7×10^3 channels over average SNR range $5 - 30$ dB for nearest neighbor and the proposed SVM classifiers with variable training set sizes.

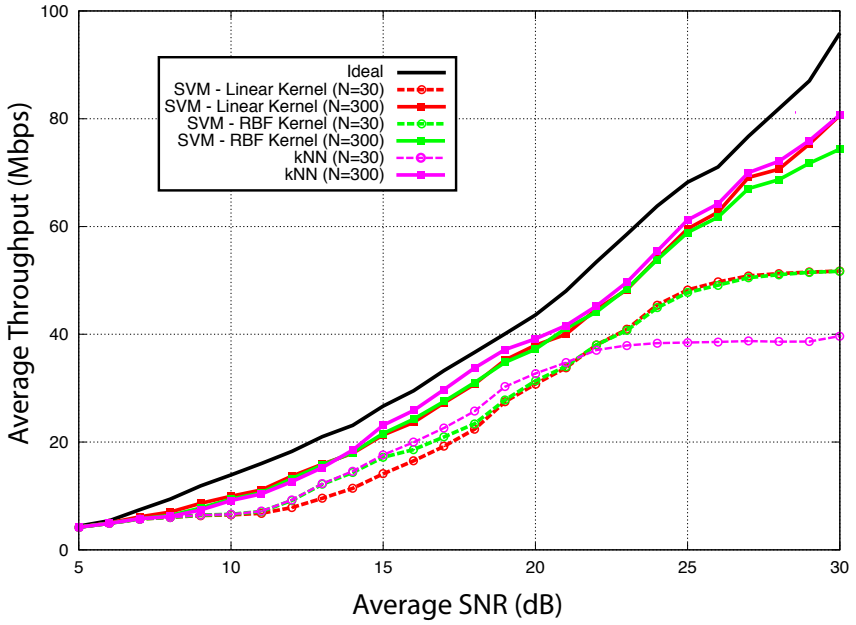


Figure 4.14: Throughput vs. binned SNR (averaged over subcarriers). Plot includes online SVM (linear and RBF kernels) and nearest neighbor for $W = 30$ and $W = 300$.

Online AMC Performance vs. Average SNR: To better understand the nature of online AMC performance, Fig. 4.14 plots the average throughput (over all channels within a binned SNR) and average FER (over all channels within a binned SNR) as a function of average SNR (channels binned to SNR regions after averaging over subcarriers). Similarly Fig. 4.15 plots the error rate averaged over SNR bins. The plots show a larger gap in throughput at high SNR and online AMC does not track the ideal FER as well. The training set I chose was disproportionately weighted against the high throughput 2×2 MCS. Essentially, even though every MCS received the same number of training samples, the larger MCS did not receive as many

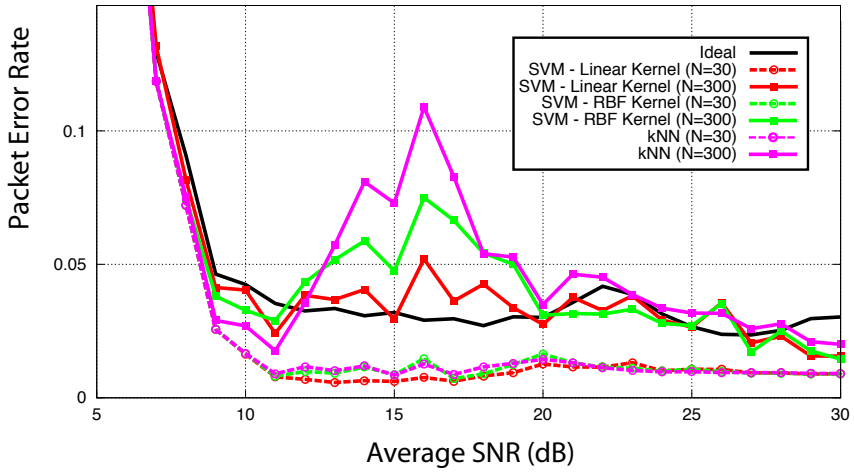


Figure 4.15: FER vs binned SNR (averaged over subcarriers). Plot includes online SVM (linear and RBF kernels) and nearest neighbor for $W = 30$ and $W = 300$.

samples near the margins. It is interesting to note that linear kernels seem sufficient in performance, and RBF kernels do not seem to be worth the extra complexity for small data set sizes.

4.4.5 Complexity Discussion

The most important characteristic of SVM online AMC, as it compares to competing algorithms, is the complexity of implementation. Complexity in the initial SVM quadratic optimization can be tolerated since it does not result in latency. Hence, I am primarily concerned with the complexity of prediction for each of the $|\mathcal{J}|$ one-versus-none classifiers and the memory storage requirements for each classifier. In the proposed SVM algorithm, prediction requires p multiplies and $p + 1$ add operations for the inference stage. The

regression stage requires 1 multiply, 2 adds, 1 division, and 1 exponential map operation. Finally, determination of the best MCS requires 1 add and 1 multiply for each class and $\lceil \log_2 |\mathcal{J}| \rceil$ compare operations between classes (assuming no uniqueness issues). In terms of memory complexity I must store the singular vectors of each of $|\mathcal{J}|$ classifiers along with the bias terms and the sigmoid map constants. I assume b-bit resolution for each of these quantities. Nearest neighbor, for the combined inference and regression stage requires W distance computations (which results in $2pW$ adds and $2pW$ multiplies) and $\log_2(W)$ compare operations. Nearest neighbor is also wasteful in terms of memory since it maintains all training data even when far from the margin. Table 5.3 summarizes the memory and processing complexity in the simulations. Clearly SVM with linear kernels relieves a huge complexity burden.

4.5 Nonlinearities and Non-Gaussian Additive Noise

Throughout this chapter I have claimed that an online AMC algorithm through supervised learning is able to capture nonlinear and non-Gaussian additive noise effects without the need for explicit modeling. Online AMC through supervised learning allows link adaptation to discover mappings between link quality metrics (feature sets) on-the-fly, while offline link adaptation is limited to the generative model used to establish link quality metric mappings. To validate these claims I run IEEE 802.11n simulations with AMC completed through the offline supervised learning algorithm developed in Chapter 3 and the online database learning algorithm developed in this

Algorithm	**Processing**	**Memory**
SVM (linear)	$\lvert\mathcal{I}\rvert \times (p+2)$ mults $\lvert\mathcal{I}\rvert$ divs $\lvert\mathcal{I}\rvert \times (p+3)$ adds $\lvert\mathcal{I}\rvert$ exp maps $\lvert\mathcal{I}\rvert \times \log_2 \lvert\mathcal{I}\rvert$ compares	$\lvert\mathcal{I}\rvert \times \mathsf{b} \times 11$
SVM (RBF)	$\lvert\mathcal{I}\rvert \times (p+2)$ mults $\lvert\mathcal{I}\rvert$ divs $\lvert\mathcal{I}\rvert \times (p+3)$ adds $\lvert\mathcal{I}\rvert$ exp maps $\lvert\mathcal{I}\rvert \times \log_2 \lvert\mathcal{I}\rvert$ compares	$\lvert\mathcal{I}\rvert \times \mathsf{b} \times 303$
kNN ($k = 10$)	$\lvert\mathcal{I}\rvert \times 600p$ mults $\lvert\mathcal{I}\rvert \times 600p$ adds $\lvert\mathcal{I}\rvert \times 9$ compares	$\lvert\mathcal{I}\rvert \times \mathsf{b} \times 300$

Table 4.3: Relevant processing/memory complexity summary for online AMC, Specific complexity numbers extracted from algorithms that achieve 31 Mbps average throughput ($W = 60$ for linear kernels, $W = 300$ for RBF kerneles, and $W = 300$ for nearest neighbor). Note that with linear kernels, 14% of training data becomes support vectors and with RBF kernels 100% of training data becomes support vectors.

chapter. The simulation parameters are summarized as follows.

- 20 MHz channels
- 2 receive antennas, 2 transmit antennas
- MCS_0 through MCS_{15} (1×1 and 2×2)
- 10% FER constraint ($\mathsf{F} = 0.1$)
- 128 B packet length
- perfect RX synchronization and channel estimation
- zero-forcing equalization
- online AMC with database learning nearest neighbor classifiers ($k = 10$)
- offline AMC with nearest neighbor classifiers trained without power amplifier nonlinearity and assuming AWGN
- both online and offline trained with 10000 channels total
- ordered post-processing SNR feature sets
- Rapp model for solid state power amplifier nonlinearity (Rapp parameter = 3 with 2 dB amplifier backoff [99])
- Laplacian additive noise model

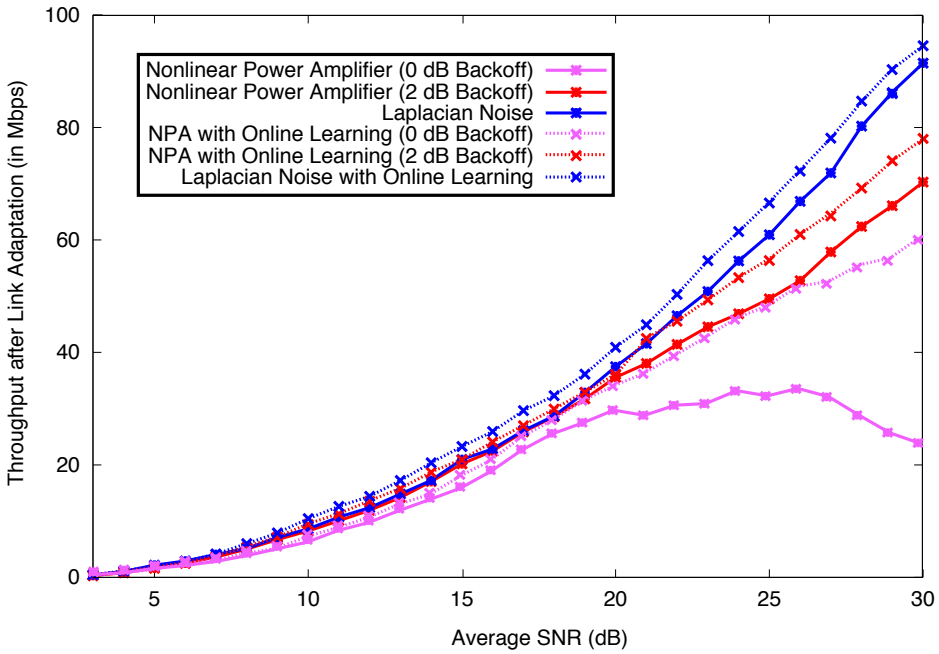


Figure 4.16: Performance comparison of IEEE 802.11n AMC with online and offline learning in spatially uncorrelated frequency flat wireless channels. Online learning improves the performance of adaptation in the presence of power amplifier nonlinearity and Laplacian additive noise. Average SNR is based on large scale path loss, not the small scale channel realization, which is statistically independent.

Figures 4.16 and 4.17 show the throughput results from this simulation for frequency flat and frequency selective channels, respectively. Clearly we see that online AMC with supervised learning is able to capture the nonlinear and non-Gaussian noise effects. In the frequency-flat channels the largest gap in performance does not occur at the highest SNR or the lowest SNR, but instead in the range between 20-30 dB. In this region many different MCSs may be selected and as a consequence more errors often occur in link

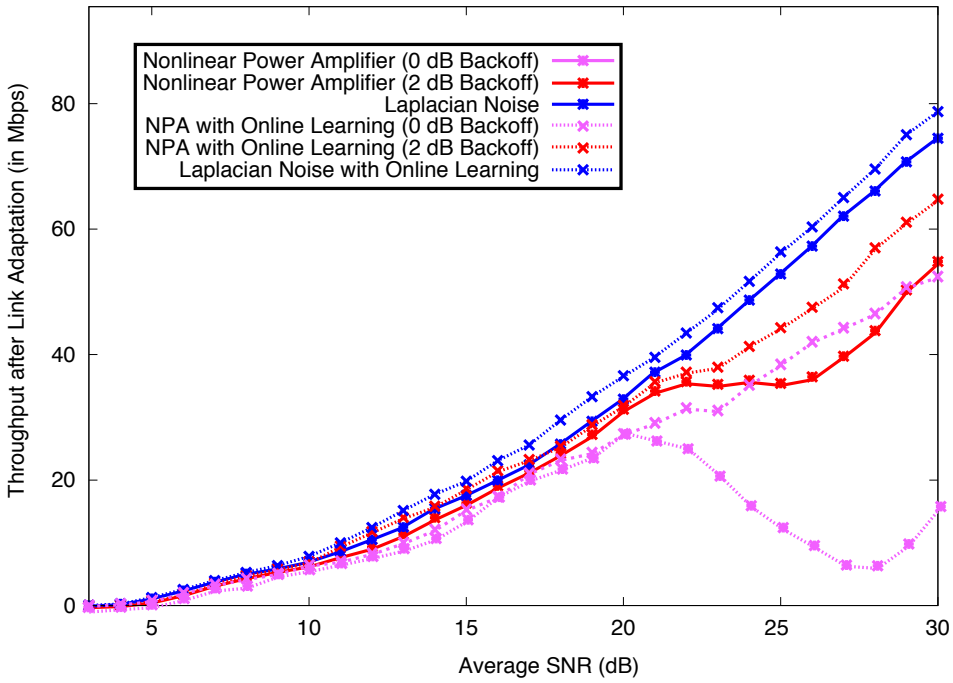


Figure 4.17: Performance comparison of IEEE 802.11n AMC with online and offline learning in spatially uncorrelated 4-tap wireless channels with uniform power delay profiles. Online learning improves the performance of adaptation in the presence of power amplifier nonlinearity and Laplacian additive noise. Average SNR is based on large scale path loss, not the small scale channel realization, which is statistically independent.

adaptation. Hence it is not surprising that offline learning performs poorly here since there are many cases where the lack of accounting for nonlinearities and non-Gaussian noise result in incorrect MCS selection. In frequency selective channels the same effects are observed, but the gap is widened dramatically when nonlinearities are observed. This suggests that amplifier backoff is more critical in frequency selective channels than it is for frequency flat channels. Hence, it may be possible through online AMC with supervised learning to

optimize amplifier backoff as a function of the wireless channel.

4.6 Summary

In this chapter I proposed the first online AMC algorithms through supervised learning. First, I developed online AMC by capturing real-time data from received frames and storing this information in databases. Nearest neighbor classifiers search through the databases and extract posterior probability class estimates which approximate FER for each MCS. To preserve diversity in the databases and maximize the performance of online AMC, I proposed a density bounding procedure. Further, I proposed exploration procedures that detect conservative link adaptation or low network utilization to prevent uninformed databases. To reduce complexity I proposed alternative classification through support vector machines. To enable posterior probability estimates for each MCS, discriminant functions are replaced with regression functions on Platt sigmoids. IEEE 802.11n simulations with SVMs show that the performance is virtually identical to nearest neighbor. The processing and memory complexity for online AMC with SVMs, however, is drastically reduced when compared to database learning with nearest neighbor classifiers.

Chapter 5

Link Adaptation in MIMO-OFDM with Non-uniform Spatial Streams

5.1 Introduction

In Chapters 3 and 4 I considered MIMO-OFDM links that feature the same constellation over all spatial streams. Current standards support this system model. For example, single-codeword MIMO in 3GPP-LTE and the mandatory MIMO modes in IEEE 802.11n all feature a single constellation for all spatial streams [17, 43]. Although frequency selectivity acts to balance the channel quality per spatial stream when open-loop precoding is used, many channel realizations in practice display a large disparity between the quality of each spatial stream. For example frequency flat channels, channels with significant spatial correlation, and channels resulting from ordered precoding all exhibit post-processing SNR that is highly variable per spatial stream [28, 78]. Consequently a throughput gain can be expected by varying the constellation per spatial stream.

Non-uniform constellation selection per spatial stream is in a nascent stage in current MIMO-OFDM standards. Developers of 3GPP-LTE recognized the throughput benefit that non-uniform constellations offered [83], but

currently non-uniform constellations are only available with multi-codeword MIMO. Multi-codeword MIMO uses a separate channel code for all streams with different constellations. This simplifies the link adaptation procedure, since AMC is easily decoupled into the system models of Chapters 3 and 4 for each codeword. Unfortunately, this architecture does not scale well for high dimension MIMO since a separate coding operation is required for each stream with a variable constellation. Currently, LTE only supports up to two different codewords, leaving up to two additional spatial streams with constellations that are not optimized for the experienced channel. To enable a architecture for varying the constellation per spatial stream that scales well with current and future high dimension MIMO systems, variable constellations with a single channel encoder are desired. This observation was also recognized by the developers of IEEE 802.11n, even though non-uniform constellations with a single encoder are currently only provided as an optional transmission mode [43].

In this chapter I study AMC in MIMO-OFDM links with a single channel encoder and non-uniform constellations per spatial stream. Explicitly, I investigate the complexity issues that have discouraged developers of wireless standards to include non-unform constellations per spatial stream. AMC is studied in the context of supervised learning classification algorithms, but the concepts apply to all AMC algorithms. The contributions of this chapter show that not only are low-complexity algorithms for non-uniform spatial streams available, but also *the total complexity of link adaptation in MIMO-OFDM*

systems can be reduced by including non-uniform constellations.

5.1.1 Contributions

Complexity reduction for AMC in MIMO-OFDM links with non-unifom spatial streams can be broken down into two categories: link quality metric dimensionality and MCS search complexity. In this chapter I show that the dimensionality of link quality metrics scales superlinearly with the number of spatial streams that offer non-uniform constellations. Moreover, I show that the search complexity increases exponentially with the number of potential constellations.

First, I address the dimensionality of link quality metrics by evaluating the structure of ordered post-processing SNR per spatial stream. I discover the connection between ordered post-processing SNR per spatial stream and the inverse marginal CDF. This leads to a new closed-form approximation for the inverse marginal CDF of ordered eigenvalues in Wishart random matrices, which, in turn, leads to the new closed form approximation for post-processing SNR in MIMO-OFDM systems with SVD precoding and asymptotic frequency selectivity. Further, by evaluating the correlation properties of OFDM subcarriers, I show that the degree with which the ordered subcarrier SNR matches the inverse CDF depends on the severity of the frequency selectivity.

Next, I construct a post-processing SNR model for variable frequency selectivity through principal component analysis (PCA). By leveraging the earlier observation of post-processing SNR convergence to the inverse marginal

CDF I am able to improve the performance of PCA. To validate the accuracy of the PCA-inspired ordered SNR model and to display its utility, I conduct Monte Carlo simulations and numerical experiments of link adaptation in an IEEE 802.11n system through supervised learning. Results of the Monte Carlo simulations show that the new model maintains accuracy regardless of the degree of frequency-selectivity or the structure of the power delay profile for the wireless channel. The link adaptation experiments demonstrate the merit of the parameters in the proposed model as link quality metrics in MIMO-OFDM systems and also demonstrates that only 2 dimensions per spatial stream are required for satisfactory link adaptation performance.

To address the aforementioned link adaptation concerns with non-uniform constellations over spatial streams, I propose and analyze the performance of a link adaptation algorithm that decouples the spatial streams. Through decoupling, *the complexity is lower than the original uniform system model*. To better understand the degree of performance sacrifice with stream decoupling, I create a bound on the maximum throughput difference between decoupled and joint stream link adaptation. I discover that, given an adequate selection of constellation orders and encoder configurations, non-uniform constellation selection over each spatial stream *shrinks the performance gap between joint and decoupled algorithms*. Numerical experiments of link adaptation through supervised learning in an uncoded non-uniform MIMO-OFDM system confirm the utility of decoupled link adaptation algorithms. Additional simulations of an IEEE 802.11n system with non-uniform spatial streams dis-

Table 5.1: Important chapter specific notation

	Description
N	number of data subcarriers per OFDM symbol
N_t	number of RF chains in the transmitter
N_r	number of RF chains in the receiver
N_s	number of spatial streams ($\leq \min\{N_t, N_r\}$)
N_d	maximum number of spatial streams ($\min\{N_t, N_r\}$)
$\mathbf{H}[n]$	$N_r \times N_t$ subcarrier channel, $n \in \{0, 1, \ldots, N-1\}$
$\mathbf{H}[\ell]$	$N_r \times N_t$ delay tap channel, $\ell \in \{0, 1, \ldots, \nu\}$
$\gamma[a, n]$	post-processing SNR of stream a, subcarrier n
$\tilde{\gamma}[a, n]$	nth smallest post-processing SNR on stream a
$f_{\mathbf{X}}(x)$	PDF of $\mathbf{X}$, i.e. $\int_{y_1}^{y_2} f_{\mathbf{X}}(x)dx = \Pr[y_1 \leq \mathbf{X} \leq y_2]$
$F_{\mathbf{X}}(x)$	CDF of $\mathbf{X}$, i.e. $F_{\mathbf{X}}(x) = \Pr[\mathbf{X} \leq x]$
$G_{\mathbf{X}}(x)$	inverse CDF of $\mathbf{X}$, i.e. $G_{\mathbf{X}}(F_{\mathbf{X}}(x)) = x$
$\lambda_a[n]$	ath unordered eigenvalue of Wishart sample $\mathbf{W}[n]$
$\tilde{\lambda}_a[n]$	ath largest eigenvalue of Wishart sample $\mathbf{W}[n]$
$\mathsf{W}[x]$	second branch of Lambert function, $x = \mathsf{W}[x]e^{\mathsf{W}[x]}$
$\mathsf{u}[x]$	unit step function, $\mathsf{u}[x] = 1, x \geq 0$ and 0 otherwise
$\mathsf{GCF}[r, s]$	greatest common factor of integers r and s
ψ_k	kth coefficient of power delay profile
p	link quality metric/feature set dimensionality per spatial stream

play the effects of frequency selectivity with variable coding rates and suggest that future standards should include a larger availability of non-uniform constellations/coding rates per stream, since it is critical towards maximizing the data rate while reducing the complexity of link adaptation procedures.

5.1.2 Important Chapter Notation

In addition to the notation presented in the system model of Section 2.3, this chapter uses the notation in Table 5.1.

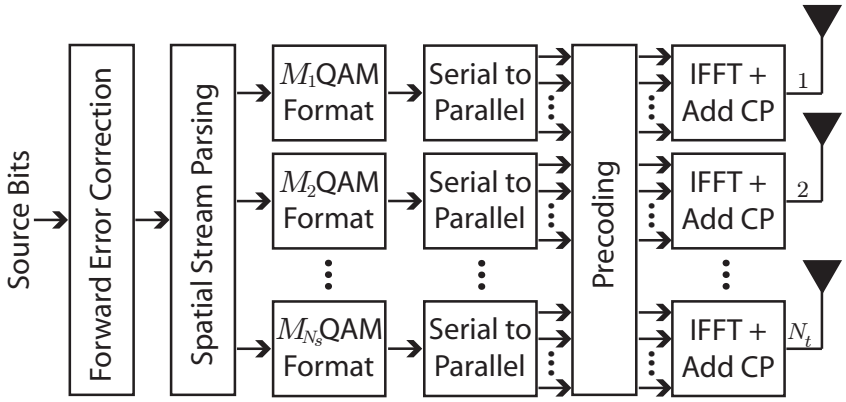


Figure 5.1: MIMO-OFDM transmission procedure where the source bits are separated into N_s spatial streams and each bit stream a is mapped K_a bits at a time into QAM constellations of order $M_a = 2^{K_a}$. Each subcarrier is precoded over the N_s streams before transmission on N_t RF chains. All interleaving is assumed to be processed on a bit-level in the spatial parsing block.

5.2 Reformatting the MIMO-OFDM System Model

In this chapter I consider a relaxed MIMO-OFDM system model where the constellation can be adapted per spatial stream. MIMO-OFDM transmit processing for each OFDM symbol is illustrated in Fig. 5.1 where N_t is the number of available RF chains at the transmitter and N_s is the number of spatial multiplexing streams (simultaneous data streams) used during transmission. For MIMO-OFDM symbol $m \in \{0, 1, \ldots, N_{\mathrm{O}} - 1\}$ and subcarrier $n \in \{0, 1, \ldots, N - 1\}$, $\mathbf{X}[m, n] \in \mathbb{C}^{N_s}$ is the transmit symbol vector. The data symbol for the ath spatial stream, $\mathsf{X}_a[m, n] \in \mathbb{C}$ for $a \in \{1, 2, \ldots, N_s\}$, represents $K_a = \log_2 M_a$ encoded bits through, for example, quadrature amplitude modulation (QAM) constellations. All other properties of the system model are unchanged including the post-processing SNR, which is defined in the same

manner as the former MIMO-OFDM symbol model through (2.22).

5.3 Link Adaptation Complexity Concerns

Although assigning non-uniform constellations per spatial stream can offer throughput gains where there exists a disparity in channel quality per spatial stream, link adaptation becomes significantly more complex. For example consider a MIMO-OFDM link with the capability to support up to N_d spatial streams and N_M potential constellations per spatial stream. Assuming complete flexibility in the choice of modulation for a single coding rate, the number of potential MCS combinations over all spatial streams for a single coding rate is

$$\sum_{k=1}^{N_d} N_M^k \tag{5.1}$$

compared to $N_d N_M$ combinations when a single MCS is chosen over all spatial streams. Hence, MIMO-OFDM links with non-uniform spatial streams experience a significant complexity increase in link adaptation with supervised learning since many more MCS options are available. To limit the search complexity, standards could restrict the number of potential constellation candidates per spatial stream if they offer equivalent performance. As I will show in this chapter, however, this may not be desirable since it limits the ability of link adaptation to decouple link adaptation per spatial stream.

The second complexity concern is the linear scaling of link quality dimensionality with the number of spatial streams. Because each stream has

a potentially different constellation, link quality metrics can no longer exploit the equivalence of spatial streams and must construct link quality metrics on a per-stream basis. Hence, the aggregate feature set dimensionality for link adaptation with supervised learning in MIMO-OFDM systems with non-uniform spatial streams grows superlinearly with the number of spatial streams. This leads to a complexity increase in the memory requirements for AMC classifiers. Moreover, the performance of supervised learning link adaptation is expected to suffer due to the curse of dimensionality.

5.4 The Ordered SNR in N-tap Gaussian Channels

In this section, to alleviate this burden on memory required for MIMO-OFDM link adaptation with non-uniform constellations per stream, I investigate the statisticals characteristics of the ordered SNR per spatial stream. As I will show in this section, the ordering operation imposes a structure that can be described in terms of the marginal statistics of post-processing SNR. The post processing SNR from (2.22) $\forall a, n$ characterizes the symbol error rate at the receiver. Chapter 3 has shown that, with adequate interleaving, the sorted post-processing SNR (i.e. ordered SNR) may also be used to accurately approximate symbol error rate. The sorting process creates many equivalence classes in the raw post-processing SNR domain. For example, any permutation of the raw post-processing SNR profile $\{\gamma[a,k]\}_{k=0}^{N-1}$ for spatial stream a is mapped to the same ordered post-processing SNR profile. Moreover, the sorted post-processing SNR sequence is non-decreasing. From these observations it

is apparent that significant structure exists in the ordered post-processing SNR profile. This structure was exploited through heuristics in Chapter 3 to compress ordered post-processing SNR into low dimension MIMO-OFDM link quality metrics, leading to more accurate link adaptation performance. In this section and the sequel I intend to discover fundamental properties of this structure, including the relationship between the ordered SNR and its empirical/marginal cumulative distribution function (CDF).

I define the $\tilde{n}$th smallest post-processing SNR value in the ath spatial stream as $\tilde{\gamma}[a, \tilde{n}] \in \{\gamma[a, 0], \gamma[a, 1], \ldots, \gamma[a, N-1]\}$. This establishes the relationship

$$\tilde{\gamma}[a, 0] \leq \tilde{\gamma}[a, 1] \leq \cdots \leq \tilde{\gamma}[a, N-1] \tag{5.2}$$

for $a \in \{1, 2, \ldots, N_s\}$. Per-stream ordering is used here, unlike in Chapter 3, so that I can allow more general adaptation models where the rate per stream is also varied.

5.4.1 Inverse Marginal CDF Approximation with SNR Ordering

Consider $\tilde{\gamma}[a, \tilde{n}]$ for a MIMO-OFDM system with channel statistics provided by subcarrier channel matrices each independently generated from the same distribution (i.e., i.i.d. subcarrier channel matrices). I use these assumptions to establish the relationship between the ordered post-processing SNR and the empirical/marginal CDF of post-processing SNR. Later, in Section 5.4.2 I relate these assumptions to channels generated from discrete time channel impulse responses.

If Γ_a is the random variable that provides realizations of $\gamma[a,n]$ in the ath spatial stream $\forall n \in \{0,1,\ldots,N-1\}$, the Glivenko-Cantelli theorem states

$$\Pr\left[\Gamma_a \leq \gamma[a,n]\right] \approx \frac{\tilde{n}}{N} \tag{5.3}$$

where $n \mapsto \tilde{n}$ in the sorting process (i.e., $\gamma[a,n] = \tilde{\gamma}[a,\tilde{n}]$) and $\tilde{n}/N$ is defined as the empirical CDF. Formally,

$$\sup_{\gamma[a,n]\in\mathbb{R}} \left| \frac{\tilde{n}}{N} - \Pr\left[\Gamma_a \leq \gamma[a,n]\right] \right| \overset{\text{a.s.}}{\to} 0 \tag{5.4}$$

as $N \to \infty$ [84, 86]. Further, it has been shown that

$$\Pr\left[\sup \left| \frac{\tilde{n}}{N} - \Pr\left[\Gamma_a \leq \gamma[a,n]\right] \right| \geq x \right] \leq Ce^{-2Nx^2} \tag{5.5}$$

for some finite constant $C \in \mathbb{R}$. An explicit value for C has also been constructed where if $Nx^2 \geq \log(2)/2$ then (5.5) is true in the strict sense for $C = 2$, i.e., $\Pr[\sup|\tilde{n}/N - \Pr[\Gamma_a \leq \gamma[a,n]]| > x] \leq 2e^{-2Nx^2}$ [64]. Consequently, if $G(y) \triangleq \inf\{x \in \mathbb{R} : y \leq \Pr[\Gamma_a \leq x]\}$, then as a consequence of Glivenko-Cantelli and the continuous mapping theorem

$$\sup \left| G\left(\frac{\tilde{n}}{N}\right) - \gamma[a,n] \right| \overset{\text{a.s.}}{\to} 0 \text{ as } N \to \infty \tag{5.6}$$

as long as $G(\cdot)$, the inverse CDF for Γ_a, only has discontinuity points of probability 0 in $(0,1)$. This motivates a new sorted subcarrier SNR model as a function of n for stream a. Further,

$$\Pr\left[\sup \left| G\left(\frac{\tilde{n}}{N}\right) - \gamma[a,n] \right| \geq x \right] \leq Ce^{-2N(x/D)^2} \tag{5.7}$$

where $D = \sup_{x \in (0,1)} G'(x)$. Intuitively, equation (5.7) suggests that the rate at which inverse mapping of the empirical CDF approaches the ordered SNR profile, in relation to the empirical CDF rate of convergence to the marginal CDF, depends on the steepness of the marginal CDF. Next, I will see that it is possible to derive a closed form approximation of $G(\cdot)$ under further system assumptions.

5.4.2 Ordered Eigenvalues

Let the channel in the discrete-time domain between receive antenna k and transmit antenna l be characterized by an N-tap time-invariant impulse response $\{[\mathbf{H}[\ell]]_{k,l}\}_{\ell=0}^{N-1}$ where $\mathbf{H}[\ell]_{k,l} \sim \mathcal{NC}(\mathbf{0}, \sigma_{\mathbf{H}}^2 \mathbf{I}/N)$. By definition, the DFT is a linear combination of the discrete-time domain samples with unity weights, $[\mathbf{H}[n]]_{k,l} = \sum_{\ell=0}^{N-1} \mathbf{H}[\ell]_{k,l} \exp\{-j2\pi\ell n/N\} \Rightarrow [\mathbf{H}[n]]_{k,l} \sim \mathcal{NC}(0, \sigma_{\mathbf{H}}^2)$ $\forall k \in \{1, 2, \ldots, N_r\}$, $\forall l \in \{1, 2, \ldots, N_t\}$. If channel information is available at the transmitter, capacity-optimal singular value decomposition (SVD), assuming optimal N_s is selected by evaluating the performance waterfillling over all possible values of N_s, may be enabled by setting $\mathbf{F}_{\text{SVD}}[n] = [\mathbf{\Phi}[n]]_{1:N_t,1:N_s} \times \mathbf{P}[n]$ and $\mathbf{G}_{\text{SVD}}[n] = \mathbf{P}^{-1}[n] \times [\mathbf{\Theta}^*[n]]_{1:N_r,1:N_s}$ where $\mathbf{\Theta}[n]\mathbf{\Sigma}[n]\mathbf{\Phi}^*[n]$ is the SVD of the channel matrix $\mathbf{H}[n]$ at subcarrier n and $\mathbf{P}[n]$ is a $N_s \times N_s$ diagonal matrix composed of the water-filling power values for each stream at subcarrier n [92]. In practice, however, power allocation over subcarriers is often not considered due to the overhead incurred compared to the gain offered in coded OFDM systems and the sensitivity to inaccurate channel state information

[4, 85]. For the remainder of this work, I will only consider SVD precoding with $\sqrt{N_s}\mathbf{P}[n] = \sqrt{E_s}\mathbf{I}_{N_s}\ \forall n$. Therefore, the post processing SNR from (2.22) can be expressed in terms of the singular values for each channel matrix, i.e. $\gamma_{\text{SVD}}[a,n] = E_s s_a[n]^2/(N_s\sigma^2)$ where $s_a[n] \triangleq [\mathbf{\Sigma}[n]]_{a,a}$.

Further, assume that the columns of $\mathbf{\Theta}[n]$ and rows of $\mathbf{\Sigma}[n]$, $\mathbf{\Phi}[n]$ are permuted to allow $s_1[n] \geq s_2[n] \geq \cdots \geq s_{N_s}[n]$. From (2.22), given a uniform power delay profile and N complex Gaussian taps, the energy-normalized sorted subcarrier SNR profile on the ath spatial stream is approximated by the inverse marginal CDF for the ath largest squared singular value of an $N_r \times N_t$ complex Gaussian random matrix. This assumes, without loss of generality, $\sigma^2_{\mathbf{H}} = \sigma^2 = 1$.

The statistics of squared singular values in complex Gaussian random matrices, or equivalently the eigenvalues in Wishart random matrices, is a well studied topic in random matrices. For i.i.d. complex Gaussian $\mathbf{H}[n]$ let

$$\mathbf{W}[n] \triangleq \begin{cases} \mathbf{H}[n]\mathbf{H}^*[n] & \text{for} \quad N_r < N_t \\ \mathbf{H}^*[n]\mathbf{H}[n] & \text{for} \quad N_r \geq N_t \end{cases} \tag{5.8}$$

which is a $\rho \times \rho$ random Wishart matrix where $\rho = \min\{N_t, N_r\}$. Let $\mathbf{\Lambda}[n] = [\lambda_1[n], \lambda_2[n], \ldots, \lambda_\rho[n]]$ be the vector of unordered eigenvalues in $\mathbf{W}[n]$. The joint probability density function (PDF) of $\mathbf{\Lambda}[n]$ is

$$f_{\mathbf{\Lambda}[n]}\left(\lambda_1[n], \lambda_2[n], \ldots, \lambda_\rho[n]\right) = \frac{1}{\rho!} \times$$
$$\prod_{k=1}^{\rho} \frac{(\lambda_k[n])^{\eta-\rho} e^{-\tilde{\lambda}_k[n]}}{(\eta-k)!(\rho-k)!} \prod_{1 \leq k < k' \leq \rho} \left(\lambda_k[n] - \lambda_{k'}[n]\right)^2 \tag{5.9}$$

where $\eta \triangleq \max\{N_r, N_t\}$ [45]. The marginal CDF of $\tilde{\lambda}_a[n]$ is computed through a ρ-fold integral. Alternative computationally efficient algorithms are also available to compute the CDF of $\tilde{\lambda}_a[n]$ [51]. Unfortunately, a simple and invertible closed form representation of the exact marginal CDF is not available.

The distribution of the extreme eigenvalues, however, can be stated in simple, closed forms. In particular, the general form of the marginal CDF for the minimum eigenvalue in $N_r \times N_t$ complex central Wishart random matrices is

$$F_{\tilde{\lambda}_\rho[n]}(x) = C + r_0 e^{-\beta_0 x} \sum_{k=0}^{\rho(\eta-\rho)} r_k x^k \tag{5.10}$$

for scalars r_k ($k \in \{0, 1, \ldots, \rho(\eta - \rho)\}$), β_0, and C calculated by taking the integral of the PDF in [106]. Note that the polynomial factor becomes trivial with symmetric links where $N_r = N_t$, resulting in the exponential distribution. The maximum eigenvalue has a similar closed form expression, but its polynomial term does not become trivial. This leaves a need to find a simple approximation for $F_{\tilde{\lambda}_a[n]}$ for all a except $a = \rho$ when $N_r = N_t$.

Recently it was observed that the SNR distribution of a single-input, multiple-output (SIMO) system with maximum ratio combining (MRC) and $L(a) \triangleq (N_t - a) \times (N_r - a)$ diversity branches approximates the SNR distribution of the ath stream in a MIMO system with SVD precoding [91]. Using the known SNR CDF of SIMO systems in Rayleigh fading [55], this approximation is defined

$$F_{\tilde{\lambda}_a[n]}(x) \approx 1 - e^{-\alpha_a x} \sum_{k=0}^{L(a)-1} \frac{\alpha_a^k}{k!} x^k. \tag{5.11}$$

given $\alpha_a \triangleq L(a)/\int_0^\infty x f_{\tilde{\lambda}_a[n]}(x)dx$ as the mean energy of the ath eigenvalue normalized by $L(a)$. This approximation has been verified to be especially precise for practical antennas dimensions (up to 10×10). Despite the simplicity of (5.11), a closed form expression of its inverse function is not available when $L(a) > 2$.

I propose a new CDF approximation that eliminates the polynomial terms of order > 1. This is equivalent to eliminating diversity branches in the SIMO system, leading to a corresponding loss in array and diversity gain. I cannot recover diversity without adding complexity to the expression, however, I can correct for the lost array gain through an appropriate shift on the support, Δ_a, and through scaling of the exponential-polynomial product term by μ_a. the new CDF approximation is defined

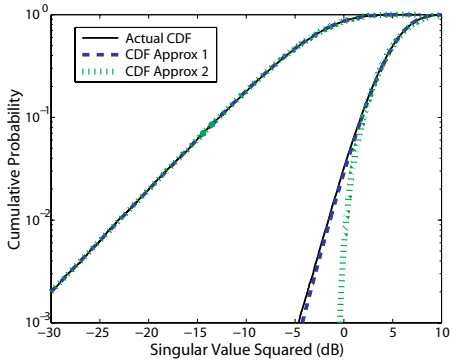
$$F_{\tilde{\lambda}_a[n]}(x) \approx 1 - \mu_a e^{-\alpha_a(x+\Delta_a)}\left(1 + \alpha_a\left(x + \Delta_a\right)\right) \tag{5.12}$$

where μ_a scales the derivative of (5.12) to match $f_{\tilde{\lambda}_a[n]}(0.5)$ and Δ_a is chosen to align (center) the approximation at $F_{\tilde{\lambda}_a[n]}(0.5)$. This produces two equations for the two unknowns, i.e.,

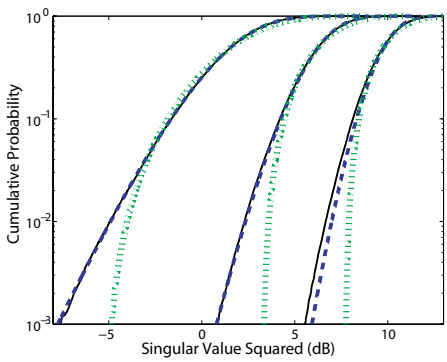
$$\mu_a = \frac{1}{\alpha_a^2(x_{0.5} + \Delta_a)} e^{\alpha_a(x_{0.5}+\Delta_a)} f_{\tilde{\lambda}_a[n]}(0.5) \tag{5.13}$$

$$0.5 = \mu_a e^{-\alpha_a(x_{0.5}+\Delta_a)}\left(1 + \alpha_a(x_{0.5} + \Delta_a)\right) \tag{5.14}$$

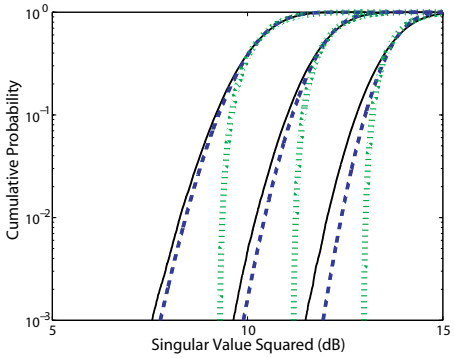
for $F_{\tilde{\lambda}_a[n]}(x_{0.5}) = 0.5$. Fig. 5.2 and Table 5.2 show the accuracy of this new approximation for the first, second, and third strongest singular values squared of 4×4, 4×8, and 8×8 complex Gaussian matrices. The approximation



(a) Ordered eigenvalues (2×2)



(b) Frst 3 ordered eigenvalues (4×4)



(c) First 3 ordered eigenvalues (8×8)

Figure 5.2: True CDF, approx. 1 from (5.11), and proposed approx. 2 from (5.12) of ordered eigenvalues in Wishart matrices.

Table 5.2: Maximum absolute error in CDF approximation (5.12)

	Max Error in $5-95\%$ **of CDF**		
Dimension	$\tilde{\lambda}_1[n]$	$\tilde{\lambda}_2[n]$	$\tilde{\lambda}_3[n]$
2×2	0.6 dB	< 0.1 dB	N/A
4×4	0.5 dB	0.5 dB	0.7 dB
8×8	0.6 dB	0.6 dB	0.6 dB
	Max Error in $1-99\%$ **of CDF**		
Dimension	$\tilde{\lambda}_1[n]$	$\tilde{\lambda}_2[n]$	$\tilde{\lambda}_3[n]$
2×2	1.8 dB	< 0.1 dB	N/A
4×4	1.3 dB	1.2 dB	0.9 dB
8×8	1.0 dB	1.0 dB	1.1 dB

results in lost accuracy in the tails of the distribution. Fortunately, the coded bit-error-rate (BER) for most practical MIMO-OFDM configurations is not largely affected by the SNR extrema, especially the larger SNR values due to the logarithmic relationship between SNR and capacity. Thus, the accuracy characterized in Fig. 5.2 and Table 5.2 is adequate for most MIMO-OFDM systems. I can also improve on this approximation by using a piecewise combination with the single exponential term derived in [32] (which is also invertible) for the left tails of the distribution. I continue without this piecewise approximation for simplicity of presentation.

To determine a continuous-domain approximation of the sorted subcarrier SNR as a function of discrete subcarriers, I need to compute the CDF inverse, $G_{\tilde{\lambda}_a[n]}$. I find that the inverse function of (5.12) is

$$G_{\tilde{\lambda}_a[n]}(x) \approx \frac{-1 - \mathsf{W}\left[\mu_a e^{-1}\left(x-1\right)\right]}{\alpha_a} - \Delta_a \tag{5.15}$$

where the notation $\mathsf{W}[x]$ indicates the value of x that solves the product of a linear term and exponential term (i.e., $x = \mathsf{W}[x]e^{\mathsf{W}[x]}$).[1] Finally, I arrive at the approximation for ordered SNR in MIMO-OFDM systems with asymptotically large frequency selectivity and SVD precoding

$$\tilde{\gamma}[a,\tilde{n}] \approx \frac{-1 - \mathsf{W}\left[\mu_a e^{-1}\left(\frac{\tilde{n}}{N}-1\right)\right]}{\alpha_a} - \Delta_a \tag{5.16}$$

for all $a \in \{1,2,\ldots,N_s\}, N_r \in \mathbb{N}, N_t \in \mathbb{N}$. Further, when $a = \rho$ and $N_r = N_t$,

[1]Here $\mathsf{W}[x] \in \mathbb{R}$ is commonly named the Lambert function or product log. It is only defined on the interval $x \in [-1/e, \infty)$ and double valued on the interval $x \in (-1/e, 0)$. Otherwise it is a non-decreasing, single-valued function.

an exact and invertible expression for the CDF is known, leading to a better ordered SNR approximation $\tilde{\gamma}[\rho, n] \approx 1/\beta_0 \log[1 - \beta_0 n/(r_0 N)]$ (see (5.10)).

5.5 Modeling Ordered Subcarrier SNR

In Section 5.4 I derived a closed form approximation of the ordered SNR for SVD precoding with asymptotically large frequency selectivity. The distance between practical ordered SNR realizations and this inverse CDF model, exponentially decays as guaranteed in (5.5) with the number of statistically independent subcarriers. In this section I investigate the subcarrier correlation as a function of the power delay profile and delay spread assuming spatially independent channels (i.e., no spatial correlation). Further, I build a general model for ordered SNR based on principal components which are established through empirical evaluation of the correlation structure in post-processing SNR. This model allows for any channel distribution, making it applicable to any transceiver configuration compatible with the system model defined in Section 5.2.

5.5.1 Subcarrier Correlation and the Power Delay Profile

The linear, time-invariant impulse response is statistically modeled in the time domain through a power delay profile (PDP). The PDP of the discrete-time impulse response between the kth receive antenna and the lth transmit antenna is denoted $[\psi_0, \psi_1, \ldots, \psi_\nu]$ such that

$$\psi_\ell = \mathbb{E}\left[[\mathbf{H}[\ell]]_{k,l} \, [\mathbf{H}[\ell]]^*_{k,l} \right] \tag{5.17}$$

where the power delay profile is the same $\forall k \in \{1, 2, \ldots, N_r\}, \forall l \in \{1, 2, \ldots, N_t\}$. If I assume, by convention, that the channel taps are i.i.d. and sampled from the zero-mean complex Gaussian distribution with variance equal to the power delay profile, the correlation coefficient between subcarriers is represented as [50]

$$\mathbb{E}\left[\left[\mathbf{H}[n]\right]_{k,l}\left[\mathbf{H}[n-\delta]\right]_{k,l}^{*}\right] = \sum_{\ell=0}^{\nu} \psi_\ell e^{-j2\pi\ell\delta/N} \tag{5.18}$$

where I constrain, without loss of generality, $\delta \in \{0, 1, \ldots, n\} \forall n \in \{0, 1, \ldots, N-1\}$. The most severe frequency selectivity occurs with a uniform power delay profile, i.e. $\psi_\ell = \psi \in \mathbb{R}\ \forall \ell \in \{0, 1, \ldots, \nu\}$, because no single channel tap dominates and all tap delays contribute equally to intersymbol interference.

Proposition 5.5.1. *For channels with a uniform power delay profile, subcarrier indices n and $n - \delta$ are statistically independent if*

$$\delta(\nu + 1)/N \in \mathbb{N}.$$

Proof. Let $\delta = qr$ and $N = rs$ where $q, r, s \in \mathbb{N}$ such that $r = \mathsf{GCF}(N, \delta)$. If $(\nu + 1) = tu$ where $t \triangleq \mathsf{GCF}(\nu + 1, s)$ for $t, u \in \mathbb{N}$, then

$$\mathbb{E}\left[\left[\mathbf{H}[n]\right]_{k,l}\left[\mathbf{H}[n-\delta]\right]_{k,l}^{*}\right] = 0 \text{ if } \delta(\nu+1)/N = d$$

since

$$\begin{aligned}\sum_{\ell=0}^{\nu} \psi_\ell e^{-j2\pi\ell\delta/N} &= \psi \sum_{\ell=0}^{u-1} e^{-j2\pi\ell q/(s/t)} \\ &= tp \sum_{\ell=0}^{u-1} e^{-j2\pi\ell d/u} = 0.\end{aligned}$$

The last equality follows from the observation that $\{0, \mathsf{mod}(d,u), \mathsf{mod}(2d,u), \ldots, \mathsf{mod}((u-1)d,u)\}$ forms the uth order cyclic group under addition with primitive element 1 for any $d \in \mathbb{N}$. This is validated by considering that $\mathsf{mod}(\ell d,u) \neq \mathsf{mod}(\ell' d,u)$ for $\ell \neq \ell' \in \{0,1,\ldots,u-1\}$ since otherwise u and d share a common factor > 1. □

Often, uniform power delay profiles do not accurately model the channel impulse response energy observed. Next, I generalize Proposition 5.5.1 to non-uniform power delay profiles.

Proposition 5.5.2. *For channels with a non-uniform power delay profile, subcarrier indices n and $n-\delta$ are statistically independent if $\delta(\nu+1)/N \in \mathbb{N}$ and*

$$\sum_{k \in \mathsf{S}_0} \psi_k = \sum_{k \in \mathsf{S}_1} \psi_k = \cdots = \sum_{k \in \mathsf{S}_{u-1}} \psi_k \tag{5.19}$$

where $\mathsf{S}_k \subset \{\psi_0, \psi_1, \ldots, \psi_\nu\}$ is the subset of power coefficients in the power delay profile with indices equal to k modulo u.

Proof. Using similar arguments from Proposition 5.5.1 I have

$$\begin{aligned} \sum_{\ell=0}^{\nu} \psi_\ell e^{-j2\pi\ell\delta/N} &= \sum_{\ell=0}^{u-1} \sum_{k \in \mathsf{S}_\ell} \psi_k e^{-j2\pi\ell q/(s/t)} \\ &= t\psi \sum_{\ell=0}^{u-1} e^{-j2\pi\ell d/u} = 0. \end{aligned}$$

□

Propositions 5.5.1 and 5.5.2 state that two subcarriers are generated from i.i.d. samples if the distance between the subcarriers and the number of taps is large enough to provide a product of subcarrier distance and discrete-time channel impulse response length that is an integer. Additionally, I require a fair distribution of power in the power delay profile. Propositions 5.5.1 and 5.5.2 are sufficient conditions for independence, but are not proven necessary. The next proposition gives a bound on the number of independently sampled subcarriers.

Proposition 5.5.3. *For all power delay profiles, every subcarrier is statistically independent of, at most, ν other subcarriers.*

Proof. Because each subcarrier channel is a zero-mean complex Gaussian random variable before and after the DFT, conservation of entropy would yield at most $\nu + 1$ independent subcarriers (since the different transmit/receive channel pairs maintain their independence) [16]. Consider, for example, subcarrier $n = N - 1$. Clearly, if $N/(\nu + 1)$ is an integer, then $\delta \in \{N/(\nu + 1), 2N/(\nu + 1), \ldots, \nu N/(\nu+1)\}$ determines independent subcarriers from Proposition 5.5.1. □

5.5.2 Modeling through Principal Components Regression

The investigation in Section 5.4 and Section 5.5.1 provided a higher understanding of the relationship between ordered post-processing SNR, its marginal CDF, and the degree of frequency selectivity in the wireless channel.

Expressions of ordered post-processing SNR element correlation for practical values of frequency selectivity, however, have not resulted from this analysis and do not appear straightforward to obtain. For example, in equation (5.18) it is not clear how this correlation structure manifests in singular values or how it relates to the ordered SNR, i.e., how ordering translates this correlation. Consequently I turn to an empirical evaluation of correlation in the ordered SNR, namely principal component analysis.

I now construct a general model, without asymptotic assumptions on frequency selectivity, the spatial correlation of separate streams, and without the requirement of SVD precoding, for ordered subcarrier SNR. Formally, I am looking for a set of orthonormal basis vectors $\mathbf{B}[a] = [\mathbf{b}_0[a]\mathbf{b}_1[a]\cdots\mathbf{b}_{p-1}[a]] \in \mathbb{R}^{N\times p}$ with $p \ll N$ such that there exists a basis component set $\mathbf{x} \in \mathbb{R}^p$ for each $\tilde{\boldsymbol{\gamma}}[a]$, the vectorized form of $\{\tilde{\gamma}[a,n]\}_{n=0}^{N-1}$, that yields $\mathbf{B}[a]\mathbf{x} \approx \tilde{\boldsymbol{\gamma}}[a]$. In practice, I do not expect to find a low-dimension basis that *exactly* characterizes the ordered SNR per stream since I have already completed two kernel transformations, DFT-domain post-processing SNR and SNR ordering, to extract the relevant performance metrics. I do, however, expect that some ordered SNR samples may be heavily correlated.

For any distribution on the channel impulse response, to evaluate the unknown correlation structure I run K experiments yielding K observations of $\tilde{\boldsymbol{\gamma}}[a]$,

$$\mathbf{Y}[a] = \left[\tilde{\boldsymbol{\gamma}}_0[a]\tilde{\boldsymbol{\gamma}}_1[a]\cdots\tilde{\boldsymbol{\gamma}}_{K-1}[a]\right]. \tag{5.20}$$

For the remainder of this chapter I will drop the $[a]$ notation with the assump-

tion that all concepts apply equally to all streams. I quantify the correlation structure through the maximum likelihood estimate of the covariance matrix

$$\mathbf{C}_{\mathbf{Y}} = \frac{1}{K} \sum_{k=0}^{K-1} \left(\tilde{\boldsymbol{\gamma}}_k - \mathsf{mean}_{k'}(\tilde{\boldsymbol{\gamma}}_{k'})\right) \left(\tilde{\boldsymbol{\gamma}}_k - \mathsf{mean}_{k'}(\tilde{\boldsymbol{\gamma}}_{k'})\right)^* \tag{5.21}$$

where $\mathsf{mean}_{k'}(\cdot)$ represents the mean vector as averaged over all observations $k' \in \{0, 1, \ldots, K-1\}$. Define $\mathbf{C}_{\mathbf{Y}} = \mathbf{E}_{\mathbf{Y}} \mathbf{D}_{\mathbf{Y}} \mathbf{E}_{\mathbf{Y}}{}^*$ as the eigenvalue decomposition of the covariance matrix where the eigenvalues are sorted in decreasing order, i.e., $[\mathbf{D}_{\mathbf{Y}}]_{k,k} \geq [\mathbf{D}_{\mathbf{Y}}]_{k+1,k+1}$ for $k \in \{0, 1, \ldots, K-2\}$. Principal component analysis (PCA) has shown that the most interesting basis vectors, i.e., those that represent data components with the largest statistical variance, correspond to the eigenvectors in $\mathbf{E}_{\mathbf{Y}}$. Moreover, the eigenvalues (diagonal of $\mathbf{D}_{\mathbf{Y}}$) determine the relative contribution of each eigenvector to the experimental observation data [47]. Finally, I conclude that

$$\mathbf{B} = [\mathbf{E}_{\mathbf{Y}}]_{:,1:p} \,. \tag{5.22}$$

Equation (5.22) finds basis vectors through naïvely implemented PCA. One drawback of naïve PCA is that the data set components with largest variance are not always the most desirable for regression. It is widely known that the median represents the order statistic with least maximum variance [73], so to ensure that the median post-processing SNR is accurately maintained, all data components are centered at the median. Intuitively, this allows PCA to capture variance of post-processing SNR with respect to the inverse CDF of the marginal distribution. I now define the median-subtracted experimental

observation set $\mathbf{Z} = [\mathbf{z}_0 \mathbf{z}_1 \cdots \mathbf{z}_{K-1}]$ with

$$\mathbf{z}_k = \tilde{\boldsymbol{\gamma}}_k[a] - \mathsf{median}(\tilde{\boldsymbol{\gamma}}_k[a])\mathbf{e} \tag{5.23}$$

where $\mathsf{median}(\cdot)$ finds the median element and $\mathbf{e}$ is the vector of all ones.[2] Proceeding with PCA in a similar manner as before I obtain empirical covariance matrix $\mathbf{C}_{\mathbf{Z}}$ and find the orthonormal basis set $\mathbf{B} = [\mathbf{E}_{\mathbf{Z}}]_{:,1:p}$. Regression with median-subtracted PCA requires an additional dimension since the median must be maintained. This extra dimension, however, will be required for all PCA implementations since large-scale channel effects must be removed in order to apply PCA with any degree of accuracy.

5.5.3 Capturing Principal Components of Channels

In this section I assume that experiments have been completed to determine the principal component basis $\mathbf{B}$ of median subtracted post-processing SNR for each spatial stream. Any new observation of the channel is represented as a combination of the principal components through weight vector $\mathbf{x} = [x_0 x_1 \cdots x_p]^*$. Next I describe the procedure for a single spatial stream with the understanding that it must be performed for each spatial stream separately.

1. Calculate $\tilde{\boldsymbol{\gamma}}$, the ordered post-processing SNR.

2. Calculate $x_0 = \mathsf{median}(\tilde{\boldsymbol{\gamma}})$.

[2]For vectors containing an even number of elements, the two middle elements may be averaged.

3. Calculate $\mathbf{z} = \tilde{\boldsymbol{\gamma}} - x_0\mathbf{e}$.

4. Calculate $[x_1 x_2 \dots x_p]^* = (\mathbf{B}^*\mathbf{B})^{-1}\mathbf{B}^*\tilde{\boldsymbol{\gamma}}$.

Linear least squares is prescribed here in the last step to determine $\mathbf{x}$, although any parameter estimation technique may be applied. Recreating an estimate of $\tilde{\boldsymbol{\gamma}}$ from $\mathbf{x}$ is straightforward since $\mathbf{B}\left[x_1 x_2 \cdots x_p\right]^* + x_0\mathbf{e} \approx \tilde{\boldsymbol{\gamma}}$.

5.5.4 Numerical Evaluation of 4×4 MIMO-OFDM

To evaluate the performance of principal component regression I simulate a MIMO-OFDM system with $N_t = N_r = 4$ and variable frequency selectivity. Because I do not assume spatial correlation, I only evaluate the performance on a single stream since all other streams provide the same statistics. The receiver implements minimum-mean-square error (MMSE) spatial equalization per subcarrier and the transmitter implements identity matrix open-loop precoding. Each plot is the result of Monte Carlo simulations with 1000 channel realizations composed of spatially/temporally i.i.d. complex Gaussian channel taps subject to a power delay profile. OFDM is processed with $N = 64$ subcarriers.

To validate the goodness of fit in principle component regression I consider two popular regression metrics. First, I consider the sum of squared error (SSE) where, for a single observation,

$$\text{SSE} = \|\mathbf{B}\left[x_1 x_2 \cdots x_p\right]^* + x_0\mathbf{e} - \tilde{\boldsymbol{\gamma}}\|_2^2. \tag{5.24}$$

SSE gives an absolute metric of the total distance between the regression and the observed data. Second I consider the coefficient of determination, R^2, which measures the strength of the linear relationship between ordered post-processing SNR and the principal component magnitudes. It is used to observe the captured percentage of variability in the regression and is defined as

$$R^2 = 1 - \frac{\|\tilde{\gamma} - \mathsf{mean}(\tilde{\gamma})\mathbf{e}\|_2^2}{\mathrm{SSE}} \tag{5.25}$$

where $\mathsf{mean}(\cdot)$ finds the mean of a single vector [79]. Figs. 5.3 and 5.4 show SSE and R^2, respectively, for various PCA configurations with post-processing SNR and the first three principal components. The first two PCA configurations, naïve PCA and PCA with median subtraction, are processed over a single multipath delay spread (i.e., a fixed number of channel taps) with uniform power delay profiles. Naïve PCA performs significantly worse than PCA with median subtraction for all multipath configurations. This justifies the intuition that centering observations about the median allows PCA to capture variation about the inverse CDF more precisely. For both PCA configurations I observe diminishing returns in adding more principal components to the regression model. This is an important observation, because it validates PCA for regression. If I did not observe this effect it would raise doubts about the utility of high variance components to represent post-processing SNR.

It may not be practical to have a separate set of principal components for each delay spread configuration. The remaining PCA configurations use median subtraction and calculate the empirical covariance matrix over all delay

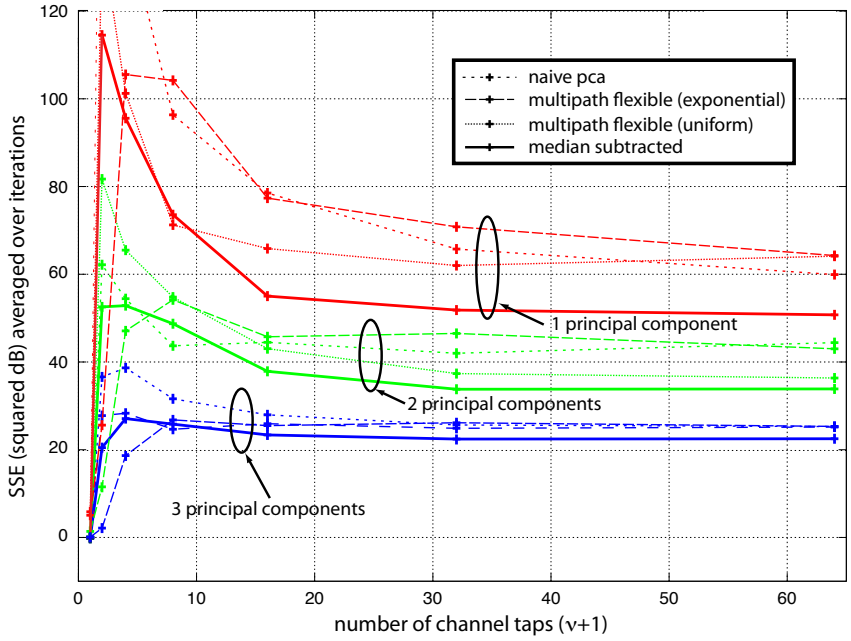


Figure 5.3: Sum of squared error (error in dB) of channel approximation with principal components in a 4×4 MIMO-OFDM system with identity matrix precoding and MMSE spatial equalization. Four different PCA configurations are considered. Multipath flexible PCA configurations are simulated with both uniform and exponential power delay profiles. The other PCA configurations are only simulated with uniform power delay profiles.

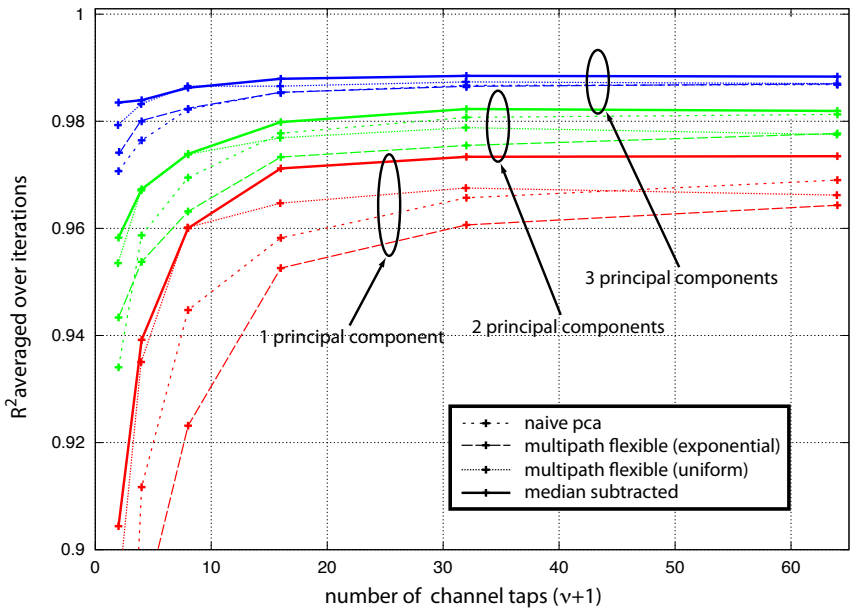


Figure 5.4: Coefficient of regression for channel approximation with principal components in a 4×4 MIMO-OFDM system with identity matrix precoding and MMSE spatial equalization. Four different PCA configurations are considered. Multipath flexible PCA configurations are simulated with both uniform and exponential power delay profiles whereas the other PCA configurations only simulated with uniform power delay profiles.

spread configurations ($\nu \in \{0, 1, 3, 7, 15, 31, 63\}$) with uniform and exponential power delay profiles (for exponential power delay profiles $\psi_\ell = \exp\{-\ell/\nu\}$). These multipath flexible PCA configurations suffer from reduced accuracy, but with 3 or more principal components, the difference is negligible. Hence, it is possible to proceed with post-processing SNR modeling through principal component regression without concerns on the sensitivity to delay spread realizations. It is notable that the performance of principal component regression is worse for exponential power delay profiles. Going back to the previous discussion in Section 5.5.1, the decaying energy of taps in exponential power delay profiles slows down the convergence to the inverse of the marginal CDF. Hence the principal statistical components do not contribute as heavily to the final ordered subcarrier SNR profile. Next, to examine this effect further, I will examine the principal components in detail.

Fig. 5.5 displays the first six principal components of the multipath flexible PCA configuration with uniform power delay profiles. Through simulation I found that both uniform and exponential power delay profiles result in very similar principal components (identical in some cases). This suggests that the covariance of median subtracted ordered post-processing SNR realizations is not largely tied to the power delay profile and results mainly from the complex Gaussian distribution of each tap. It is interesting to observe that the primary component of the post-processing SNR very closely resembles the inverse marginal CDF, which is shown as a reference through numerical calculations since no closed form is available. Also notice that because the tails of

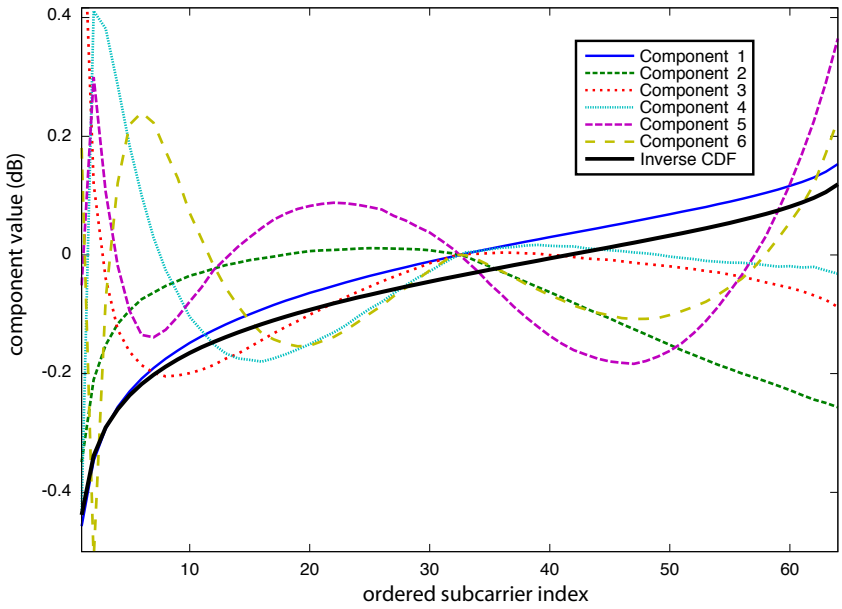


Figure 5.5: The first six principal components post-processing SNR per stream in a 4×4 MIMO-OFDM system with identity matrix precoding and MMSE spatial equalization. The principal component closely approximates the marginal CDF of post processing SNR. All components are normalized on a dB scale.

the inverse CDF provide the largest maximum variance, they are significantly represented by all of the components. Because the components were computed by evaluating post-processing SNR on a dB scale, the low-SNR tails provide the most uncertainty.

5.5.5 Link Quality Metrics

The first motivation for this study was to determine if better link quality metrics/feature sets are available for MIMO-OFDM links with non-uniform spatial streams and a single encoder realization, such that the complexity of link adaptation due to increasing memory could be reduced. A comprehensive IEEE 802.11n link adaptation simulation campaign with supervised learning, similar to that presented in Chapter 3, was completed for various power delay profiles and delay spread realizations. I computed the best known multi-dimensional link quality metrics using the procedure from Chapter 3 (strategically selected ordered post-processing values per stream) and compared their performance to the component magnitudes of PCA. While the throughput performance gain was marginal enough that the results are not presented here, the dimensionally of the feature set was further reduced through PCA coefficients. Consider Figure 5.6 which shows the number of errors in 4-,8-, and 16-tap channels without spatial correlation and with uniform power delay profiles. With PCA, I halve the dimensionality of the feature sets used in supervised learning AMC algorithms. Given the perceived error floor for MCS selection errors through supervised learning, it is unlikely that better link quality met-

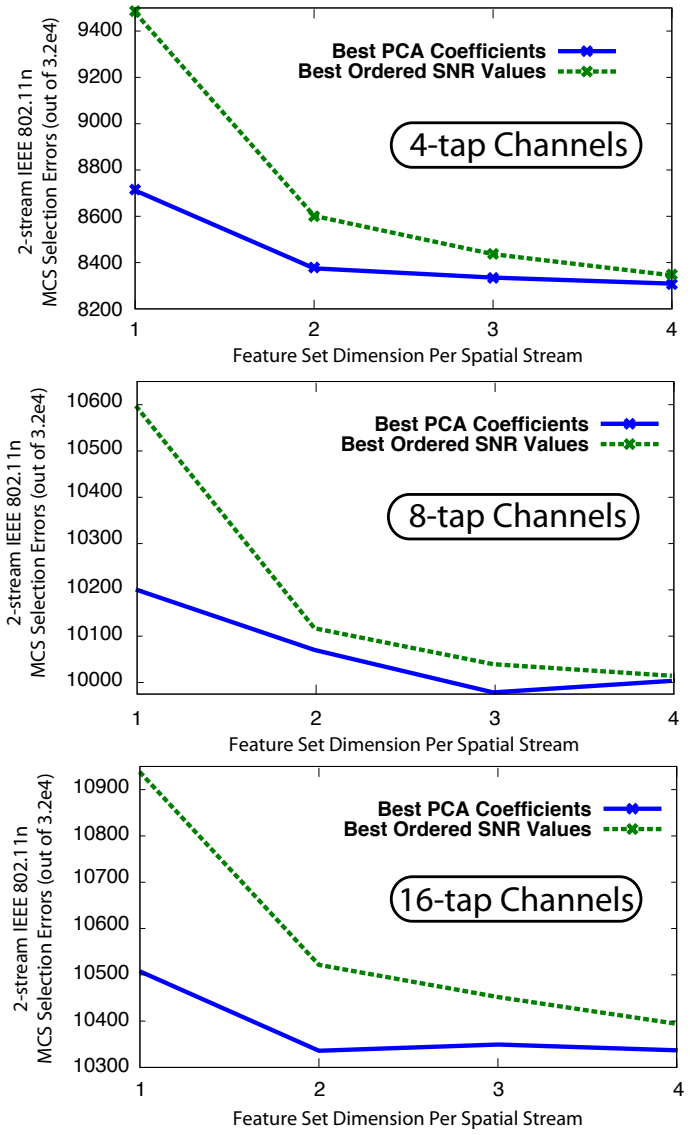


Figure 5.6: Errors in supervised learning AMC for 2×2 IEEE 802.11n MIMO-OFDM wireless links with non-uniform constellations per spatial stream (MCS_0 through MCS_{15} and MCS_{33} through MCS_{38}). Channels were spatially uncorrelated with 4-tap uniform power-delay profiles. The average SNR varied from 4 to 34 dB.

rics area available.

5.6 AMC through Supervised Learning

AMC in the reformatted MIMO-OFDM system model with non-uniform constellations per spatial stream is the process of selecting the number of spatial streams, N_s, the constellation order for spatial stream a, M_a $\forall a$, and the channel encoder configuration. In this section each realization of N_s, $\{M_1, M_2, \ldots, M_{N_s}\}$, and a single encoder configuration will represent an MCS. As considered in the previous sections of this chapter, training data for supervised learning can no longer exploit subcarrier SNR ordered over spatial streams to reduce dimensionality in the feature sets. Hence, the feature sets that represent the link quality metric for the wireless channel are an aggregate of the link quality metrics per spatial stream. The remainder of the AMC process with supervised learning is the same as discussed in Chapters 3 and 4 for offline and online algorithms, respectively.

5.6.1 Joint Stream Link Adaptation

Joint stream link adaptation is the straightforward translation of supervised learning for AMC in MIMO-OFDM systems with non-uniform constellations. Joint algorithms allow for high accuracy, but also suffer from high complexity because no commonality/correlation between each spatial stream is exploited. The algorithm is as follows.

1. Define $k := 0$.

2. Extract the $(k+1) \times p$-dimensional feature set (p dimensions for each stream) from channel state measurements and store in query $\mathbf{q}$.

3. Use classifier c_k, trained with $\{\mathbf{z}_{k,0}, \ldots, \mathbf{z}_{k,W-1}\}$ and the associated $(k+1)$-stream classes, to predict the mapping $c_k : \mathbf{q} \mapsto i(k)$ where $i(k)$ is the highest rate $(k+1)$-stream MCS meeting the reliability constraint for the current channel state.

4. Increment k and repeat steps 2-4 until $k = N_d$.

5. Select highest rate MCS in $\{i(0), \ldots, i(N_d - 1)\}$.

If $\{c_0, c_1, \ldots, c_{N_d-1}\}$ are all ideal, the algorithm performs perfect link adaptation according to the reliability constraint.

5.6.2 Decoupled Stream Link Adaptation

Joint stream link adaptation requires N_d different classifiers. To save silicon space, processing power, and battery energy I propose a link adaptation algorithm that decouples the spatial streams as illustrated in Fig. 5.7. *The key assumption of stream decoupling is that $k+1$ single stream MCSs may be mapped to an equivalent $(k+1)$-stream MCS.* For the moment, I will accept this as true and will defer the discussion of such mappings to the next subsection. The decoupled stream link adaptation algorithm is as follows.

1. Define $k := 0$. Define $a := 1$.

2. Compute the effective $(k+1)$-stream channel.

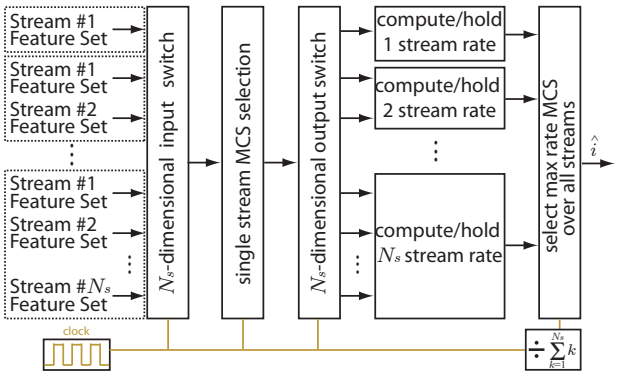


Figure 5.7: Operation of decoupled link adaptation. N_s feature sets are first extracted from the $N_s \in \{1, 2, \ldots, N_d\}$ stream channel. Each feature set is used to complete a single stream link adaptation procedure. The N_s single-stream MCSs are mapped to a single N_s stream rate. The value of N_s used for transmission is determined by the maximum rate MCS.

3. Extract the p-dimensional feature set for stream a and store it in the current query $\mathbf{q}$. Use the classifier c_0, trained with $\{\mathbf{z}_{0,0}, \mathbf{z}_{0,1}, \ldots, \mathbf{z}_{0,W-1}\}$ and the associated 1-stream classes, to complete mapping $c_0 : \mathbf{q} \mapsto i(0, a)$ where $i(0, a)$ is the highest rate 1-stream MCS meeting the reliability constraint for stream a in the current channel state.

4. Increment a and repeat step 3 until $a = k + 2$.

5. Map single stream MCS classes $\{i(0, 1), \ldots, i(0, k+1)\}$ to $(k+1)$-stream equivalent MCS $i(k + 1)$.

6. Increment k and repeat steps 2-5 until $k = N_d$.

7. Select highest rate MCS in $\{i(0), \ldots, i(N_d - 1)\}$.

Algorithm	Memory Complexity
Joint	$\mathsf{M} \times \sum_{k=1}^{N_d} k$
Decoupled	M

Table 5.3: Memory complexity comparison for joint and decoupled stream link adaptation in non-uniform MIMO-OFDM systems. M is the required memory for a classifier and feature set in a single stream. The processing complexity for decoupled link adaptation is also much less because the decoupled operation does not have to complete new searches to consider permutations of the constellation order on each stream.

I observe reduced complexity since only one classifier, c_0, is needed. The training data required is also reduced, which often significantly reduces the memory required in classifiers. A complexity comparison of joint and decoupled link adaptation for MIMO-OFDM is summarized in Table 5.3.

5.6.3 Decoupled Stream Mappings

Decoupled stream link adaptation requires a single stream MCS mapping from the channel for each of N_s spatial streams separately. The algorithm assumes that there exists a unique mapping to an N_s-stream MCS. The standard MIMO-OFDM system model, however, only allows for a single channel encoder realization over all subcarriers and spatial streams per frame. Hence, because encoder configurations (primarily coding rate) are adapted to provide different data rates as a function of channel quality, combinations of single-stream MCSs with mixed coding rates will not map to a unique N_s-stream MCS.

Therefore, for illustrative purposes, I only continue with single stream

MCSs that all have the same encoder configuration, since otherwise I cannot guarantee reliability constraints will not be violated.[3] I define the following algorithm for mapping between N_s single-stream MCSs and one N_s-stream MCS. Further I assume that the single stream MCSs are always ordered in terms of decreasing reliability for all channels, i.e., $\text{FER}_0(\cdot) \leq \text{FER}_1(\cdot) \leq \cdots \leq \text{FER}_{|\mathcal{I}_0|-1}(\cdot)$ where $|\mathcal{I}_{N_s-1}|$ is defined as the number of N_s-stream MCSs in $\{0, 1, \ldots, |\mathcal{I}| - 1\}$.

1. Compute $\{i(0,1), i(0,2), \ldots, i(0,N_s)\}$, the best single stream MCS for each stream in effective N_s-stream channel.

2. Find $\arg\max_{i_1,\ldots,i_{N_s}} \{\sum_{a=1}^{N_s} \mathcal{R}_{i_a} : i_a < i(0,a) \forall a\}$

Reliability ordering is not always valid in frequency-selective channels, but these exceptions happen infrequently in practice, especially for fixed encoder configurations [53].

5.6.4 Decoupled Link Adaptation Performance Loss

Using the assumption of frequency-flat channels, I define γ_i as the average SNR (over subcarriers) that yields $\text{FER}_i = \mathsf{F}$ for $0 \leq i < |\mathcal{I}_0|$. I will assume that any combination of N_s 1-stream MCSs maps to a only one N_s-

[3] Even with a single encoder configuration, decoupled stream may violate FER constraints in frequency-selective channels, depending on the interleaving configuration. For example, if two frequency-selective spatial streams are interleaved together in a convolutionally-coded MIMO-OFDM system, the weaker subcarriers of each spatial stream can be brought together after deinterleaving to produce higher probability error events [58].

stream MCS. I also assume that each MCS uses the same encoder configuration and different rates are achieved by changing the constellation order.

Because the number of available MCSs is finite, the adapted throughput does not change continuously with average SNR. This results in a suboptimal expected rate since any single stream average SNR, temporarily defined γ_{avg}, must be mapped to a single stream MCS_i such that $\gamma_i \leq \gamma_{\text{avg}} < \gamma_{i+1}$ to meet the per-stream FER constraint. In contrast, with joint stream link adaptation, I can balance the FER across streams and reach a higher overall throughput. Let the throughput of single-stream transmission over frequency flat channels with average SNR γ_{avg} be represented

$$T(\gamma_{\text{avg}}) \approx \sum_{i=0}^{|\mathcal{J}_0|-1} \mathcal{R}_i \left[\mathsf{u}(\gamma_{\text{avg}} - \gamma_i) - \mathsf{u}(\gamma_{\text{avg}} - \gamma_{i+1})\right] \tag{5.26}$$

for $\gamma_{|\mathcal{J}_0|} = \infty$, unit step function $\mathsf{u}(\cdot)$ and assuming that a step function approximation is suitable for the chosen FER constraint F with throughput as defined $(1 - \text{FER}) \times \text{Rate}$.

I further assume that decoupled stream link adaptation never selects a single stream MCS for any spatial stream that is more than one index below the equivalent spatial stream MCS that results from joint stream MCS selection. I justify this assumption by observing that equal shifts in SNR on a dB scale above and below an MCS boundary (10% FER) do not produce equal shifts in FER. Since the MCSs are designed to be equally distributed in SNR on a dB scale and because the rate of FER increase above the 10% FER SNR boundary

is much larger than the rate of FER decrease below the SNR boundary for typical MCS realizations, this assumption is often valid in practice [52].

Given the aforementioned system model and its assumptions, I acknowledge that the worst case scenario occurs when the average SNR for all but one of the spatial streams is just below the FER constraint boundary. In that scenario, joint stream MCS selection will balance the FER across the spatial streams by bumping up all but one of the single stream MCSs. Thus, I state that the maximum performance gap is

$$\Delta T_{\max} = (N_s - 1) \times \max_{i \in \{1,2,\ldots,|\mathcal{I}_0|-1\}} \{\mathcal{R}_i - \mathcal{R}_{i-1}\}. \tag{5.27}$$

Clearly, by decreasing the rate gap between adjacent single stream MCSs with the same channel encoder configuration, I decrease the maximum potential performance gap.

5.6.5 Simulations

I complete numerical simulations first using the IEEE 802.11n standard with 20 MHz channels, 2 receive antennas, 2 transmit antennas, and MCS_0 through MCS_{15} (uniform streams) along with MCS_{33} through MCS_{38} (non-uniform streams) [43]. Link adaptation is performed with k-nearest neighbor (k-NN) classifiers and ordered post-processing SNR feature sets according to Chapter 3. In these simulations, however, for joint stream MCS selection, since non-uniform spatial streams are considered, I cannot perform SNR ordering over the spatial streams, but instead *only order per-spatial stream.* This increases the dimensionality of the feature set in joint MCS selection by a factor

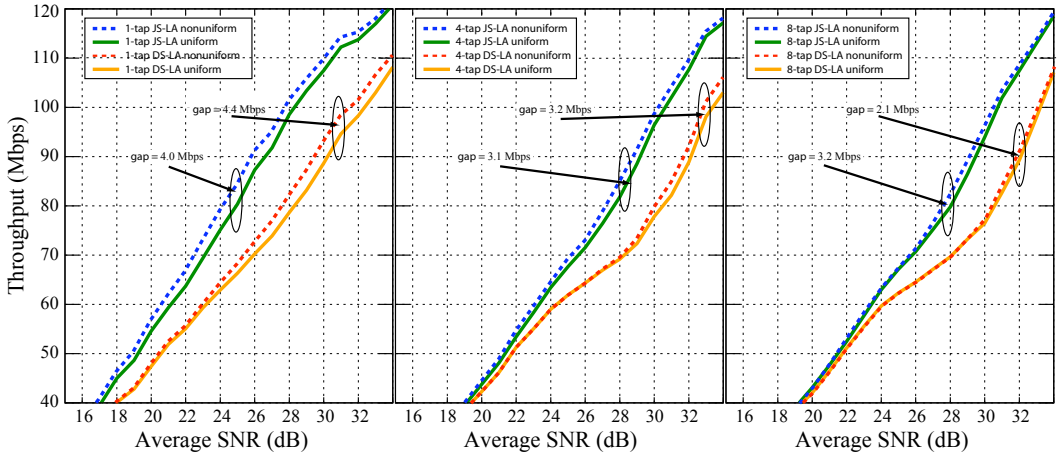


Figure 5.8: Throughput of 2×2 IEEE 802.11n systems with joint stream (JS) and decoupled stream (DS) link adaptation using k-NN classifiers. The throughput resulting from ideal link adaptation was not plotted because the curves were not significantly distinguishable (the mean/max throughput loss from k-NN link adaptation was 0.0%/0.2%,1.7%/5.3%, and 2.2%/6.7% for 1, 4, and 8 tap channels respectively). The uniform curves only adapt over MCS_0 through MCS_{15} while the non-uniform curves leverage the remaining MCS_{33} through MCS_{38}

of $N_s = 2$ when compared to the features used in Chapter 3 and decoupled stream MCS selection. Fig. 5.8 shows the throughput generated from complex Gaussian channels for 1, 4, and 8 delay taps with uniform power delay profiles. Each packet consists of 128 bytes, uses no precoding, is perfectly synchronized, and the receiver implements zero-forcing equalization with error-free channel estimation.

The gains for non-uniform modulation (max throughput gain $\approx$ 4 Mbps) are not indicative of the maximum gains in practice. SVD precoding which creates a larger disparity between stream quality and higher MIMO dimensionality which creates more potential streams. Adding selectivity to the channel increases the equality of spatial streams on average. I observe less channels where non-uniform constellations are advantageous. *To maximize the gains of non-uniform constellation selection, precoding is desired to maintain channel quality disparity among streams.* The gap ($\approx 10 - 20$ Mbps) between the performance of joint and decoupled stream link adaptation is significant, which was expected for IEEE 802.11n since there is only a small number of available mappings between single stream rates and double stream rates. These simulations encourage future MIMO-OFDM standards to consider a more comprehensive list of available coding rates and non-uniform constellations to enable high-performance, reduced-complexity link adaptation.

To further demonstrate this point a second set of supervised learning link adaptation simulations were completed, this time with BPSK, QPSK, 8-PSK, and 16-PSK modulation in 3×3 MIMO-OFDM with frequency flat

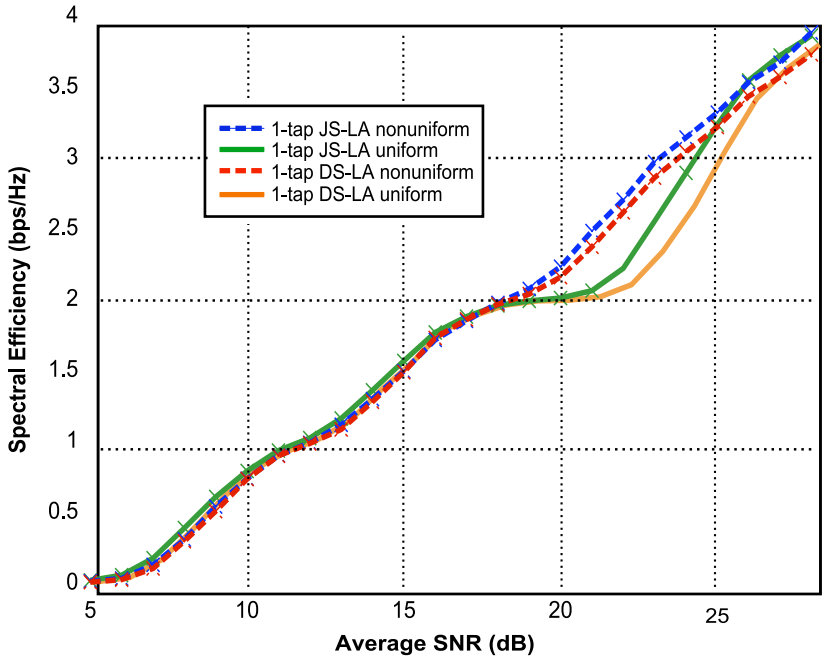


Figure 5.9: 3×3 MIMO-OFDM uncoded system performance for BPSK, QPSK, 8-PSK, and 16-PSK modulation per stream. SVD precoding is implemented to create stream disparity. Decoupled link adaptation performance approaches joint link adaptation performance for non-uniform modulation.

channels, SVD precoding, and no forward error correction. All combinations of different dimension PSK constellations are available to create non-uniform spatial streams up to $N_s = 3$. Fig. 5.9 shows the resulting throughput for k-NN classifiers and ordered post-processing SNR feature sets. I observe significant gains for non-uniform modulation ($\approx$ 0.6bps/Hz) over uniform modulation. Moreover I can demonstrate that decoupled link adaptation approaches the performance of joint link adaptation when non-uniform modulation is enabled. This confirms the intuition that non-uniform modulation can serve to both increase the overall throughput and allow low-complexity, high performance

link adaptation for channels with stream quality disparity.

5.7 Summary

In this chapter I studied complexity reduction techniques for supervised learning AMC in MIMO-OFDM links with non-uniform constellations per spatial stream and a single channel encoder. First I characterize the post-processing SNR per spatial stream in terms of the inverse marginal CDF and show convergence asymptotically. This was used to develop a closed form approximation of ordered SNR as a function of ordered subcarrier index in asymptotically frequency selective channels with SVD precoding. A more general model was developed through PCA allowing for compression of the channel state into a small number of dimensions. This compression was used to reduce the dimensionality of link quality metrics in MIMO-OFDM links with non-uniform constellations. Finally, I proposed a decoupled stream link adaptation procedure which drastically reduces the complexity of link adaptation in the sytem model of this chapter. Further I showed that if an adequate selection of constellations are available in the standard, the performance loss of decoupled link adaptation becomes negligible.

Chapter 6

Conclusion

6.1 Summary

In this dissertation link adaptation in modern MIMO-OFDM wireless systems has been improved through machine learning. A chapter-by-chapter summary of the dissertation is included as follows.

Chapter 3 provides the first design of link adaptation algorithms for coded, bit-interleaved MIMO-OFDM systems through machine learning. First, to enable improvement in the accuracy of link adaptation in MIMO-OFDM systems, I created a new link quality metric resulting from post-processing SNR ordered over subcarriers and spatial streams. This new link quality metric was inspired by a new frame-error rate bound I derived in this chapter. This bound shows that the link performance of MIMO-OFDM systems is well approximated by the post-processing SNR, independent of subcarrier and spatial stream location assuming sufficient bit interleaving. Second, I create a link adaptation algorithm based on nearest neighbor supervised learning classification that is compatible with arbitrary link quality metrics. This classification algorithm infers the operation regions of separate transmission parameters based on the offline analysis of training data. The training data includes a mul-

titude of channel realizations and the correct transmission parameter mapping for a fixed frame-error rate reliability constraint. Practical simulations with IEEE 802.11n validate the supervised learning classifier for link adaptation by outperforming the best known MIMO-OFDM link quality metrics.

In Chapter 4, I create online link adaptation algorithms with machine learning that are able to perceive shifts in the operating regions for each transmission parameter in terms of a fixed link quality metric. These shifts may be due to nonlinearities or non-Gaussian noise effects. In contrast to the offline learning algorithms, the training set for the proposed online learning link adaptation does not contain the best known transmission parameters for a fixed channel, but only contains single measurements of frame success or failure for a fixed transmission parameter, housed in a database that is updated in real-time. The first proposed online learning link adaptation algorithms are enabled by maximum likelihood estimates of the posterior probabilities in nearest neighbor classification algorithms on the training set. Multiple algorithms are proposed to maintain a diversity of training set observations in finite sized databases. Further, the database is refreshed so that link adaptation is not tainted with older training data. IEEE 802.11n Hydra prototype measurements shown that online learning link adaptation outperforms competing link adaptation algorithms in practical systems. I also propose a second online learning link adaptation algorithm, this time through support vector machines. The posterior probability estimate is extracted from SVMs through regression on the Platt sigmoid. IEEE 802.11n simulations show that the complexity of

online learning link adaptation with SVMs was nearly identical to the performance of nearest neighbor. Additionally, I show that SVMs significantly reduce the implementation complexity through appropriate selection of the kernel function and training set size.

Finally, Chapter 5 provides a framework for link adaptation in coded and bit-interleaved MIMO-OFDM systems where the digital symbol constellation varies per spatial stream. First, I analyze the statistics of ordered SNR per spatial stream in MIMO-OFDM systems and show that asymptotically (with frequency selectivity) this converges to the marginal CDF of post-processing SNR. Further, I derive a closed form approximation of the post-processing SNR in terms of the ordering index through the inverse function of an approximation of the marginal CDF of the post-processing SNR. Using this insight I leverage unsupervised learning, namely principal component analysis, to determine the primary components of ordered SNR. This leads me to a new link quality metric in terms of the principal components, resulting in increased accuracy for lower dimension link quality metrics. Unfortunately, even with this new insight, the complexity of link adaptation with non-uniform constellations is still fairly complex. To further reduce the complexity I propose a decoupled link adaptation procedure where each spatial stream is treated separately. I provide a bound on the performance of this decoupled link adaptation procedure and determine that future MIMO-OFDM wireless systems should allow for non-uniform constellations per spatial stream with a sufficient number of constellation options for each spatial stream such that the performance of

decoupled stream link adaptation approaches the joint stream algorithm.

6.2 Future Work

Several future research directions are identified below.

Reducing the Search Depth in Link Adaptation: As wireless design moves forward, the number of transmission parameter sets continues to grow. Currently, the optimal link adaptation algorithm is forced to evaluate the performance for each transmission parameter set. There is a large correlation however, between the frame error rate of one transmission parameter set and another in a fixed channel. Given a set of channels and the frame error rate for each transmission parameter set for each channel, a learning algorithm should be able to identify the candidate transmission parameter sets that are worth considering for link adaptation.

Feature Sets for Impairments: In this dissertation feature sets were designed to capture the total performance of LTI-AWGN links with MIMO-OFDM. Although online learning was able to adapt to impairments outside this model, such as device nonlinearity and non-Gaussian noise, feature sets that capture information about these model violations may lead to superior performance. For example, Volterra series parameters can be incorporated into the system model, resulting in more advanced link quality metrics. Moreover, higher dimensional statistical metrics such as skewness and kurtosis, may better characterize the additive noise. Further, measurements of network contention in random access networks can be used to predict the likelihood of

collisions.

SVM Improvements: In the proposed online learning SVM, iterative algorithms that define inference and regression functions were re-initiated when the number of elements in the training set changed. Online methods, however, that allow new training elements to be added to the ongoing optimization in real-time should be feasible based on past work [49]. It is also desirable for SVM optimization procedures that place an upper bound on the number of available support vectors to limit memory complexity. ν-SVM provides a potential framework for this, but it is not clear that ν-SVM is reliable in practice. Further, to make online AMC with SVMs completely practical, feature set diversity algorithms developed in the database learning AMC protocol need to be translated to SVMs. This might be enabled, for example, through density constraints on the support vectors chosen.

Design and Incorporation of New Switching Metrics: To take advantage of space-time/frequency codes in current supervised learning AMC algorithms, all potential configurations must be evaluated in terms of the expected SNR gain before computation of link quality metrics. This is disadvantageous because of the number of potential configurations. One alternative is to incorporate MIMO switching metrics which determine the operating regions of diversity strategies. This is complicated by the current MIMO-OFDM link model which may display variable operating regions per subcarrier and spatial stream. Hence, new switching metrics are desired to reduce link adaptation complexity in modern systems.

Power Amplifier Backoff Optimization as a Function of the Wireless Channel State: Chapter 4 showed that the degree of performance loss due to power amplifier nonlinearity in systems with online AMC through supervised learning depends on the frequency selectivity of the wireless channel. Backing off on the power amplifier can improve the linearity of the channel response, but this may not be worth the lost transmit power. Because the nonlinearity properties of the power amplifier are fundamentally linked to the exact transmitter used and the device temperature, only online AMC algorithms with supervised learning offer any hope of optimizing the backoff/throughput tradeoff as a function of the frequency and spatial selective wireless channel.

Unordered Post-Processing SNR Link Quality Metrics: Ordered post-processing SNR was leveraged in this dissertation to provide low-dimension link quality metrics. Analysis in Chapter 5 through PCA, however, suggests that a fundamental limit exists for link adaptation accuracy in MIMO-OFDM links (with bit-interleaving and error control coding) that use link quality metrics derived from ordered SNR. Since all of the best known link quality metrics remove information about the ordering of SNR (per spatial stream, per subcarrier, or both), this suggests that future link quality metrics should focus on the incorporation of unordered information. Until such link quality metrics are developed, increased accuracy in AMC for MIMO-OFDM in selective channels is not anticipated. Intuitively, this results from non-ideal interleaving and finite length frames.

Bibliography

[1] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris. Link-level measurements from an 802.11b mesh network. *SIGCOMM Comput. Commun. Rev.*, 34(4):121–132, 2004.

[2] Jeffrey G. Andrews, Arunabha Ghosh, and Rias Muhamed. *Fundamentals of WiMAX: Understanding Broadband Wireless Networking (Prentice Hall Communications Engineering and Emerging Technologies Series)*. Prentice Hall PTR, March 2007.

[3] S. Arya et al. An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *J. ACM*, 45(6):891–923, 1998.

[4] Furrer S. Barreto, A.N. Adaptive bit loading for wireless OFDM systems. In *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, volume 2, pages G–88 –G–92 vol.2, Sep/Oct 2001.

[5] Kevin S. Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. When is "nearest neighbor" meaningful? In *Proceedings of the 7th International Conference on Database Theory*, pages 217–235, London, UK, 1999. Springer-Verlag.

[6] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, August 2006.

[7] S. Bittner, E. Zimmermann, and G. Fettweis. Exploiting phase noise properties in the design of MIMO-OFDM receivers. In *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*, pages 940 –945, April 2008.

[8] Y.W. Blankenship, P.J. Sartori, B.K. Classon, V. Desai, and K.L. Baum. Link error prediction methods for multicarrier systems. *Proceedings of the IEEE Vehicular Technology Conference, Fall*, 6:4175–4179 Vol. 6, Sept. 2004.

[9] K. Brueninghaus, D. Astely, T. Salzer, S. Visuri, A. Alexiou, S. Karger, and G.-A. Seraji. Link performance models for system level simulations of broadband radio access systems. *Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 4:2306–2311 Vol. 4, Sept. 2005.

[10] J. Cavers. Variable-rate transmission for Rayleigh fading channels. *IEEE Transactions on Communications*, 20(1):15–22, Feb 1972.

[11] Chih C. Chang and Chih J. Lin. LIBSVM: a library for support vector machines. http://www.csie.ntu.edu.tw/ cjlin/libsvm, 2001.

[12] Y.-S. Choi and S. Alamouti. A pragmatic PHY abstraction technique for link adaptation and MIMO switching. *IEEE Journal on Selected*

Areas in Communications,, 26(6):960–971, August 2008.

[13] S. Choudhury and J.D. Gibson. Throughput optimization for wireless LANs in the presence of packet error rate constraints. *IEEE Communications Letters*, 12(1):11–13, January 2008.

[14] John Cioffi. *Digital Communications*. Stanford University Course Reader, May 2010.

[15] T. Claasen and A. Jongepier. Model for the power spectral density of quantization noise. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 29(4):914 – 917, Aug 1981.

[16] T. M. Cover and J. A. Thomas. *Elements of Information Theory 2nd Edition*. Wiley-Interscience, July 2006.

[17] E. Dahlman, S. Parkvall, J. Skold, and P. Beming. *3G Evolution, Second Edition: HSPA and LTE for Mobile Broadband*. Academic Press, 2008.

[18] A. A. D'Amico and M. Morelli. Joint Tx-Rx MMSE design for MIMO multicarrier systems with Tomlinson-Harashima precoding. *IEEE Transactions on Wireless Communications*, 7(8):3118–3127, August 2008.

[19] R. C. Daniels and R. W. Heath, Jr. 2x2 IEEE 802.11n link adaptation data sets. http://www.profheath.org/, 2009.

[20] R. C. Daniels and R.W. Heath, Jr. 60 GHz wireless communications: emerging requirements and design recommendations. *IEEE Vehicular Technology Magazine*, 2(3):41–50, Sept. 2007.

[21] R. C. Daniels, K. Mandke, K. T. Truong, S. M. Nettles, and R. W. Heath, Jr. Throughput/delay measurements of limited feedback beamforming in indoor wireless networks. In *Proceedings of the IEEE Global Communications Conference*, 2008.

[22] S. S. Das, F. Tariq, M. I. Rahman, F. B. Frederiksen, E. De Carvalho, and R. Prasad. Impact of nonlinear power amplifier on link adaptation algorithm of OFDM systems. In *IEEE Vehicular Technology Conference*, pages 1303 –1307, Oct 2007.

[23] A. Demir. Phase noise and timing jitter in oscillators with colored-noise sources. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 49(12):1782 – 1791, Dec 2002.

[24] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000.

[25] Ericsson. System-level evaluation of OFDM - further considerations. In *3GPP Contributions*, number 35 R1-031303. TSG-RAN WG1, November 2003.

[26] D. Falconer, S.L. Ariyavisitakul, A. Benyamin-Seeyar, and B. Eidson. Frequency domain equalization for single-carrier broadband wireless systems. *IEEE Communications Magazine*, 40(4):58–66, Apr 2002.

[27] M. Faulkner and M. Johansson. Correction of mixer nonlinearity in quadrature modulators. *Electronics Letters*, 28(3):293 –295, Jan. 1992.

[28] A. Forenza, D. J. Love, and Heath, Jr. R. W. Simplified spatial correlation models for clustered MIMO channels with different array configurations. *IEEE Transactions on Vehicular Technology*, 56(4):1924 –1934, July 2007.

[29] J. H. Friedman, J. L. Bentley, and R. A. Finkel. An algorithm for finding best matches in logarithmic expected time. *ACM Trans. Math. Softw.*, 3(3):209–226, 1977.

[30] T. Fujisawa et al. A single-chip 802.11a MAC/PHY with a 32-b RISC processor. *IEEE Journal of Solid-State Circuits*, 38(11):2001–2009, Nov. 2003.

[31] M. J. Gans. A power-spectral theory of propagation in the mobile-radio environment. *IEEE Transactions on Vehicular Technology*, 21(1):27 – 38, Feb 1972.

[32] L. Garcia-Ordonez, A. Pages-Zamora, and J. R. Fonollosa. Diversity and multiplexing tradeoff of spatial multiplexing MIMO systems with CSI. *IEEE Transactions on Information Theory*, 54(7):2959–2975, July 2008.

[33] A.J. Goldsmith and S.-G. Chua. Variable-rate variable-power MQAM for fading channels. *IEEE Transactions on Communications*, 45(10):1218–1230, Oct 1997.

[34] A.J. Goldsmith and P.P. Varaiya. Capacity of fading channels with channel side information. *IEEE Transactions on Information Theory*, 43(6):1986–1992, Nov 1997.

[35] Andrea Goldsmith. *Wireless Communications*. Cambridge University Press, August 2005.

[36] M.A. Haleem and R. Chandramouli. Adaptive stochastic iterative rate selection for wireless channels. *IEEE Communications Letters*, 8(5):292–294, May 2004.

[37] J. Halton and A. Spaulding. Error rates in differentially coherent phase systems in non-gaussian noise. *IEEE Transactions on Communication Technology*, 14(5):594 –601, Oct 1966.

[38] S. Hara, A. Ogino, M. Araki, M. Okada, and N. Morinaga. Throughput performance of SAW-ARQ protocol with adaptive packet length in mobile packet data transmission. *IEEE Transactions on Vehicular Technology*, 45(3):561 –569, Aug 1996.

[39] T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning*. Springer, August 2001.

[40] J. Hayes. Adaptive feedback communications. *IEEE Transactions on Communication Technology*, 16(1):29–34, Feb 1968.

[41] R.W. Heath, Jr. and A.J. Paulraj. Switching between diversity and multiplexing in MIMO systems. *IEEE Transactions on Communications*, 53(6):962–968, June 2005.

[42] R.W. Heath, Jr., S. Sandhu, and A. Paulraj. Antenna selection for spatial multiplexing systems with linear receivers. *IEEE Communications Letters*, 5(4):142–144, Apr 2001.

[43] IEEE 802.11n Working Group. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Enhancements for Higher Throughput*, part 11 standard edition, 2009.

[44] T. Issariyakul and E. Hossain. Performance modeling and analysis of a class of arq protocols in multi-hop wireless networks. *IEEE Transactions on Wireless Communications*, 5(12):3460 –3468, Dec 2006.

[45] A. T. James. Distributions of matrix variates and latent roots derived from normal samples. *The Annals of Mathematical Statistics*, 35(2):475–501, 1964.

[46] R. Johannesson and K. S. Zigangirov. *Fundamentals of Convolutional Coding*. Wiley-IEEE Press, 1999.

[47] I.T. Jolliffe. *Principal Component Analysis*. Springer Verlag, 2002.

[48] S. Kant and T. L. Jensen. Fast link adaptation for IEEE 802.11n. Master's thesis, Aalborg University, February 2007.

[49] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy. Improvements to Platt's SMO algorithm for SVMs classifier design. *Neural Computation*, 13(3):637–649, 2001.

[50] M. Krondorf, S. Bittner, and G. Fettweis. Numerical performance evaluation for OFDM systems affected by phase noise and channel estimation errors. In *Proceedings of the IEEE Vehicular Technology Conference*, pages 1–5, Sept. 2008.

[51] R. Kwan, C. Leung, and P. Ho. Distribution of ordered eigenvalues of Wishart matrices. *Electronics Letters*, 43(5):31–32, 1 2007.

[52] M. Lamarca and F. Rey. Indicators for PER prediction in wireless systems: A comparative study. *Proceedings of the IEEE Vehicular Technology Conference*, 2:792–796, June 2005.

[53] M. Lampe, H. Rohling, and W. Zirwas. Misunderstandings about link adaptation for frequency selective fading channels. *Proceeedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2:710–714 vol.2, 15-18 Sept. 2002.

[54] P. Lee. Computation of the bit error rate of coherent M-ary PSK with gray code bit mapping. *IEEE Transactions on Communications*, 34(5):488 – 491, May 1986.

[55] W. C.Y. Lee. *Mobile Communications Engineering: Theory and Applications*. McGraw-Hill, Inc., New York, NY, USA, 1997.

[56] J. Li and A. Stefanov. Exact pairwise error probability for block-fading MIMO OFDM systems. *IEEE Transactions on Vehicular Technology*, 57(4):2607–2611, July 2008.

[57] H. T. Lin, C. J. Lin, and R. C. Weng. A note on Platt's probabilistic outputs for support vector machines. *Machine Learning*, 68(3):267–276, Oct 2007.

[58] Shu Lin and Daniel J. Costello. *Error Control Coding, Second Edition*. Prentice Hall, April 2004.

[59] H. Liu, L. Cai, H. Yang, and D. Li. EESM based link error prediction for adaptive MIMO-OFDM system. *Proceedings of the IEEE Vehicular Technology Conference*, pages 559–563, April 2007.

[60] D. J. Love, R. W. Heath, V. K. N. Lau, D. Gesbert, B.D. Rao, and M. Andrews. An overview of limited feedback in wireless communication systems. *IEEE Journal on Selected Areas in Communications*, 26(8):1341 –1365, Oct 2008.

[61] Y. Ma. Improving wireless link delivery ratio classification with packet SNR. In *Proceedings of the IEEE International Conference on Electro Information Technology*, May 2005.

[62] Ketan Mandke, Soon-Hyeok Choi, Gibeom Kim, Robert Grant, Robert C. Daniels, Wonsoo Kim, Scott M. Nettles, and Robert W. Heath Jr. Early

Results on Hydra: A Flexible MAC/PHY Multihop Testbed. In *Proceedings of the 65th IEEE Vehicular Technology Conference*, pages 1896–1900, April 2007.

[63] R. C. Manso. Performance analysis of M-QAM with Viterbi soft-decision decoding. Master's thesis, Naval Postgraduate School, Monterey, CA, March 2003.

[64] P. Massart. The tight constant in the Dvoretzky-Kiefer-Wolfowitz inequality. *The Annals of Probability*, 18(3):1269–1283, 1990.

[65] M.R. McKay and I.B. Collings. Capacity and performance of MIMO-BICM with zero-forcing receivers. *IEEE Transactions on Communications*, 53(1):74–83, Jan. 2005.

[66] A. Misra, V. Krishnamurthy, and R. Schober. Stochastic Learning Algorithms for Adaptive Modulation. In *IEEE International Conference on Acoustics, Speech and Signal Processing*.

[67] T. M. Mitchell. *Machine Learning*. McGraw-Hill, March 1997.

[68] S. Nanda and K.M. Rege. Frame error rates for convolutional codes on fading channels and the concept of effective Eb/N0. *IEEE Transactions on Vehicular Technology*, 47(4):1245–1250, Nov 1998.

[69] A. Niculescu-Mizil and R. Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*, page 632. ACM, 2005.

[70] M. Nourani and A. Radhakrishnan. Testing on-die process variation in nanometer VLSI. *IEEE Design Test of Computers*, 23(6):438 –451, June 2006.

[71] Y. Oda, R. Tsuchihashi, K. Tsunekawa, and M. Hata. Measured path loss and multipath propagation characteristics in UHF and microwave frequency bands for urban mobile communications. volume 1, pages 337–341, 2001.

[72] I.M. Onyszchuk. Truncation length for Viterbi decoding. *IEEE Transactions on Communications*, 39(7):1023–1026, Jul 1991.

[73] N. Papadatos. Maximum variance of order statistics. *Annals of the Institute of Statistical Mathematics*, 47(1):185–193, 1995.

[74] A. Paulraj, R. Nabar, and D. Gore. *Introduction to Space-Time Wireless Communications*. Cambridge University Press, 2003.

[75] F. Peng, J. Zhang, and W. E. Ryan. Adaptive Modulation and Coding for IEEE 802.11n. In *Proceedings of the IEEE Wireless Communications and Networking Conference*, pages 656–661, Kowloon, Mar 2007.

[76] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74, 1999.

[77] Theodore S. Rappaport. *Wireless Communications: Principles and Practice (2nd Edition)*. Prentice Hall PTR, December 2001.

[78] F. Rey, M. Lamarca, and G. Vazquez. Robust power allocation algorithms for MIMO OFDM systems with imperfect CSI. *IEEE Transactions on Signal Processing*, 53(3):1070 – 1085, March 2005.

[79] P.J. Rousseeuw and A.M. Leroy. *Robust Regression and Outlier Detection*. Wiley-IEEE, 2003.

[80] B.C. Sarkar. On the joint statistics of amplitude and phase of a signal with co-channel interference. *Proceedings of the IEEE*, 76(3):298 –299, mar 1988.

[81] T. C. W. Schenk, R. W. van der Hofstad, E. R. Fledderus, and P. F. M. Smulders. Distribution of the ICI term in phase noise impaired OFDM systems. *IEEE Transactions on Wireless Communications*, 6(4):1488–1500, April 2007.

[82] D. Seethaler, G. Matz, and F. Hlawatsch. An efficient MMSE-based demodulator for MIMO bit-interleaved coded modulation. *IEEE Global Telecommunications Conference*, 4:2455–2459, Dec. 2004.

[83] Freescale Semiconductor. EUTRA downlink MIMO configurations and comparisons. In *3GPP TSG RAN WG1 #43, R1-060248*, 2006.

[84] Robert J. Serfling. *Approximation Theorems of Mathematical Statistics*. Wiley-Interscience, February 1981.

[85] A. Seyedi and G.J. Saulnier. Robust bit-loaded wireless OFDM. In *Proceedings of the IEEE Vehicular Technology Conference*, volume 1, pages 593 – 597, Sept 2004.

[86] G.R. Shorack and J.A. Wellner. *Empirical processes with applications to statistics*. Wiley New York, 1986.

[87] S. Simoens, S. Rouquette-Léveil, P. Sartori, Y. Blankenship, and B Classon. Error prediction for adaptive modulation and coding in multiple-antenna OFDM systems. *Signal Processing*, 86(8):1911–1919, 2006.

[88] C. K. Sung, S.-Y. Chung, J. Heo, and I. Lee. Adaptive bit-interleaved coded OFDM with reduced feedback information. *IEEE Transactions on Communications*, 55(9):1649–1655, Sept. 2007.

[89] L. Szczecinski, C. Gonzalez, and S. Aissa. Exact expression for the BER of rectangular QAM with arbitrary constellation mapping. *IEEE Transactions on Communications*, 54(3):389 – 392, March 2006.

[90] P.H. Tan, Y. Wu, and S. Sun. Link adaptation based on adaptive modulation and coding for multiple-antenna OFDM system. *IEEE Journal on Selected Areas in Communications*, 26(8):1599–1606, October 2008.

[91] T. Taniguchi, S. Sha, and Y. Karasawa. Analysis and approximation of statistical distribution of eigenvalues in i.i.d. MIMO channels under Rayleigh fading. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Science*, 91(10):2808–2817, 2008.

[92] Emre Telatar. Capacity of multi-antenna gaussian channels. *European Transactions on Telecommunications*, 10:585–595, 1999.

[93] TGn Channel Models Special Committee. TGn channel models. IEEE 802.11-03/940r1, November 2003.

[94] S. Tsai and A. Soong. Effective-SNR mapping for modeling frame error rates in multiple-state channels. In *3GPP Standardization Document*, number 3GPP-C30-20030429-010. 3GPP2, 2003.

[95] A. Tsertou and D. I. Laurenson. Insights into the hidden node problem. In *Proceedings of the International Conference on Wireless Communications and Mobile Computing*, pages 767–772, New York, NY, USA, 2006. ACM.

[96] A. Viterbi. Convolutional codes and their performance in communication systems. *IEEE Transactions on Communications*, 19(5):751–772, Oct 1971.

[97] P.K. Vitthaladevuni, M.-S. Alouini, and J.C. Kieffer. Exact BER computation for cross QAM constellations. *IEEE Transactions on Wireless Communications*, 4(6):3039 – 3050, Nov. 2005.

[98] W.T. Webb and R. Steele. Variable rate QAM for mobile radio. *IEEE Transactions on Communications*, 43(7):2223–2230, Jul 1995.

[99] G.P. White, A.G. Burr, and T. Javornik. Modelling of nonlinear distortion in broadband fixed wireless access systems. *Electronics Letters*, 39(8):686 – 687, April 2003.

[100] N. Wolf, J.-E. Mueller, and H. Klar. Simple predistortion system for compensation of temperature dependent nonlinearity of power amplifiers. In *Proceedings of the IEEE Radio and Wireless Symposium (RWS)*, pages 152 –155, Jan 2010.

[101] G. Xu and Y. Lu. Channel and modulation selection based on support vector machines for cognitive radio. In *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing*, pages 1–4, Wuhan, Sept. 2006.

[102] J. H. Yeong, X. M. Rao, M. R. Shajan, Q. Wang, J. C. Y. Lin, and X. H. Qu. 802.11a MAC layer: firmware/hardware co-design. In *International Conference on Information, Communications and Signal Processing*, volume 3, pages 1923–1928 vol.3, Dec. 2003.

[103] K. Yu and B. Ottersten. Models for MIMO propagation channels: a review. *Wireless Communications and Mobile Computing*, 2(7):653–666, 2002.

[104] On-Ching Yue. Design trade-offs in cellular/PCS systems. *IEEE Communications Magazine*, 34(9):146 –152, Sep 1996.

[105] S. Yun and C.M. Caramanis. Multiclass support vector machines for adaptation in MIMO-OFDM wireless systems. In *Proceedings of the Allerton Conference on Communication, Control, and Computing*, 2009.

[106] H. Zhang, F. Niu, H. Yang, and D. Yang. Polynomial PDF of the smallest eigenvalue of complex central Wishart matrix with correlation at the side with the smallest number of antennas. *IEEE Communications Letters*, 12(10):749–751, Oct. 2008.

[107] L. Zheng and D. Tse. Diversity and multiplexing: a fundamental tradeoff in multiple-antenna channels. *IEEE Transactions on Information Theory*, 49(5):1073–1096, 2003.

[108] Q. Zou, M. Mikhemar, and A.H. Sayed. Digital compensation of cross-modulation distortion in software-defined radios. *IEEE Journal of Selected Topics in Signal Processing*, 3(3):348 –361, June 2009.

Vita

Robert Clark Daniels was born and raised in Tunkhannock, Pensylvania. He received Bachelor of Science degrees in Electrical Engineering and Mathematics from The Pennsylvania State University in 2004. In 2006 he received the Masters of Science in Engineering degree in Electrical and Computer Engineering from The University of Texas at Austin. He is currently the Executive Vice President of Engineering at Kuma Signals, LLC in Austin, TX where he serves as Principal Investigator on multiple Department of Defense contracts. His expertise includes wireless physical layer algorithm implementation, 60 GHz wireless link design, multiple antenna communications, link adaptation, and the application of machine learning algorithms to wireless networks.

Permanent address: 11903A Doubloon Cv
Austin, Texas 78759

This dissertation was typeset with LaTeX[†] by the author.

[†]LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's TeX Program.